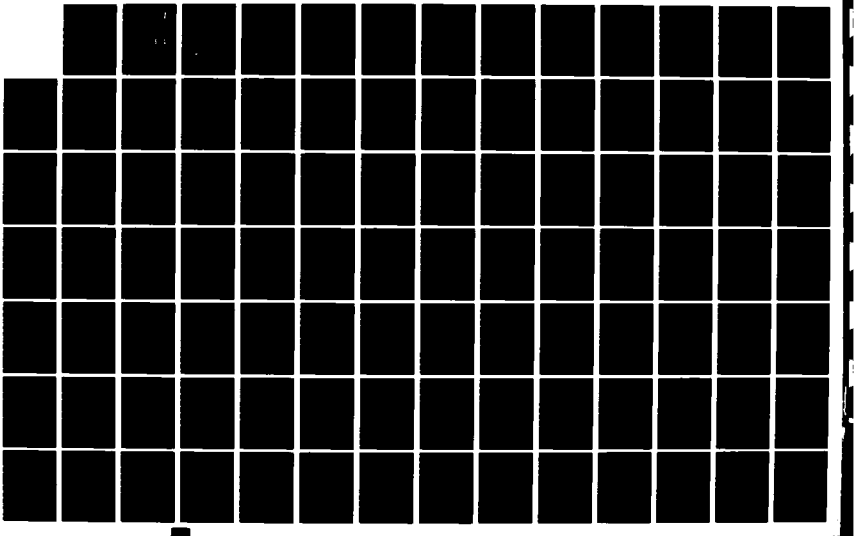
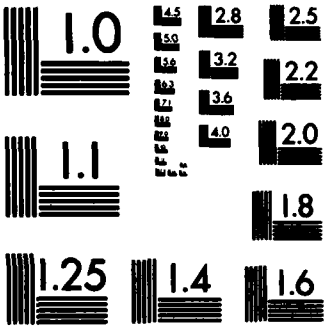
AD-A133 510 DESIGN AND INTEGRATION SUPPORT TO FLIGHT INTERFACE HARDWARE(U) QUEST RESEARCH CORP MCLEAN VA R B NORMOYLE 30 APR 82 QRC-C-4136 N00014-81-C-2499 1/2

UNCLASSIFIED F/G 9/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

PHOTOGRAPH THIS SHEET

AD A133510

DTIC ACCESSION NUMBER

II

LEVEL

INVENTORY

Rpt. No. C-4136, Final, 30Apr.'82

DOCUMENT IDENTIFICATION

Normoyle, Robt. B.;
Contract N00014-81-C-2499

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

DISTRIBUTION STATEMENT

ACCESSION FOR	
NTIS	GRA&I
DTIC	TAB
UNANNOUNCED	
JUSTIFICATION	
BY	
DISTRIBUTION /	
AVAILABILITY CODES	
DIST	AVAIL AND/OR SPECIAL
A	

DISTRIBUTION STAMP

DTIC
ELECTE
S OCT 13 1983 D
D

DATE ACCESSIONED

83 10 12 302

DATE RECEIVED IN DTIC

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-DDA-2

DTIC FORM OCT 79 70A

DOCUMENT PROCESSING SHEET

AD A133 510

DESIGN AND INTEGRATION SUPPORT TO FLIGHT INTERFACE HARDWARE

QUEST RESEARCH CORPORATION

6858 Old Dominion Drive
McLean, Virginia 22101
(703) 821-3200

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

DESIGN AND INTEGRATION SUPPORT TO FLIGHT INTERFACE HARDWARE

Submitted to:

NRL
Code 5706
Washington, D.C. 20375

559,401

Submitted by:

Quest Research Corporation
6858 Old Dominion Drive
McLean, Virginia 22101
(703) 821-3200

C-4136

30 April 1982

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

THIS PAGE INTENTIONALLY LEFT BLANK

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) DESIGN AND INTEGRATION SUPPORT TO FLIGHT INTERFACE HARDWARE		5. TYPE OF REPORT & PERIOD COVERED FINAL 16 Jul 81 - 31 Mar 82
		6. PERFORMING ORG. REPORT NUMBER C-4136
7. AUTHOR(s) Robert B. Normoyle		8. CONTRACT OR GRANT NUMBER(s) N00014-81-C-2499
9. PERFORMING ORGANIZATION NAME AND ADDRESS QUEST RESEARCH CORPORATION 6858 Old Dominion Drive McLean, VA 22101		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS NAVAL RESEARCH LABORATORY Tactical Electronic Warfare Division (5606) Washington, DC 20375		12. REPORT DATE 30 April 1982 13. NUMBER OF PAGES 97
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Central Target Simulator
Network Design

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Design and integration support to flight test hardware was provided to the Naval Research Laboratory during the period 16 July 81 to 31 March 82. Computer hardware and software were developed to perform digital and analog input/output to a missile seeker. A PDP 11/23 and the necessary software were designed, installed, and tested to perform input/output and computer-to-computer communications. Methods of automatically booting a remote computer through another computer were developed and implemented. Software was written for a terminal switching network that would permit the connection of any CTS computer terminal to any of 16 system computers.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

FOREWORD

The effort described in this report was performed for the Naval Research Laboratory, 4555 Overlook Avenue S.W., Washington, DC 20375. Quest Research Corporation, 6858 Old Dominion Drive, McLean, VA 22101 performed the technical effort from 16 July 1981 to 31 March 1982 under Contract N00014-81-C-2499. Contractor engineer was Robert B. Normoyle. The Naval Research Laboratory Contracting Officer Technical Representative (COTR) was Mr. Robert Surratt.

During the period of performance, three major tasks were completed:

- Installation of Harpoon seeker computer software and hardware.
- Testing of changes in ERP effects of target position.
- Feasibility of on-line graphics processor.

TABLE OF CONTENTS

SECTION		PAGE
	FOREWORD	
1.0	**INSTALLATION OF HARPOON SEEKER COMPUTER SOFTWARE AND HARDWARE**	1-1
1.1	Hardware Implementation and Evaluation of Computer Boards	1-1
1.2	Feasibility of Using Previously Written Software for the Communications Link	1-1
1.3	Creation of a more Efficient Software Package	1-3
1.4	Program and Network Debugging	1-3
1.5	Creation and Implementation of an Automatic Boot	1-3
1.6	Conversion of the New PDP 11/23 Software to Run on the SMI Seeker Computer	1-4
2.0	**TESTING OF CHANGES IN ERP EFFECTS ON TARGET POSITION**	2-1
3.0	**FEASIBILITY OF ON-LINE COLOR GRAPHICS PROCESSOR**	3-1
APPENDICES		
A.	DLH23	A-1
B.	H23	B-1
C.	BOOT	C-1
D.	BOOT23	D-1
E.	COMM2	E-1
F.	TASM	F-1
G.	TUDUM2.MAC	G-1
H.	FTCRBN	H-1
I.	FTC Results	I-1

1.0 INSTALLATION OF HARPOON SEEKER COMPUTER SOFTWARE AND HARDWARE

The task where most support was concentrated was adapting a PDP 11/23 computer to act as a communications link between missiles and a network of computers. This support may be separated into six categories:

a. Hardware implementation and evaluation of computer boards.

b. Feasibility of using previously written software for the communications link.

c. Creation of a more efficient software package.

d. Program and network debugging.

e. Creation and implementation of an automatic boot.

f. Conversion of the new 11/23 software to run on the SM1 seeker computer.

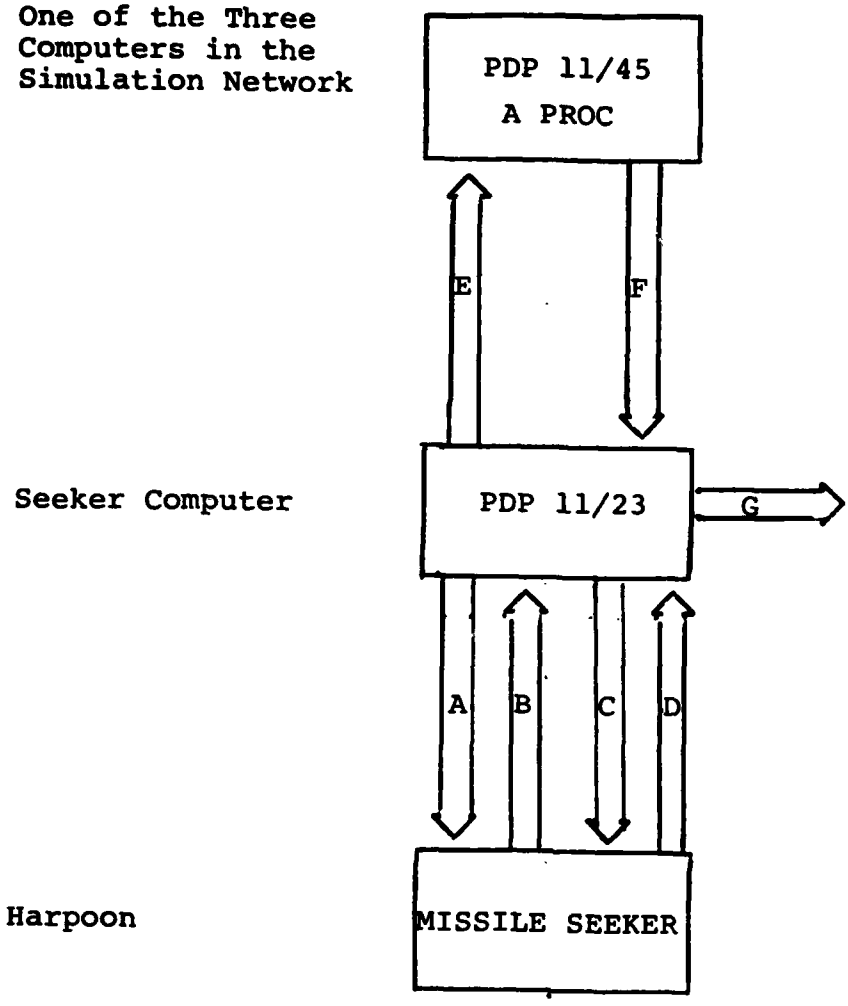
1.1 Hardware Implementation and Evaluation of the Computer Boards

The PDP 11/23 consisted of an LSI 11/23 CPU, analog-to-digital (A/D) converters, digital-to-analog (D/A) converters, and serial and parallel input/output (I/O) devices. Each of these boards was completely tested to assure that they met Navy specifications. Calibration data was collected for the A/D and the D/A converters. The accuracy of these devices was determined to be within desired specifications. The computer I/O devices were then connected to the instruments and the communications links were tested. The instruments were then tested to assure that they react properly to various signals.

At present all hardware and communication links are capable of performing the designated tasks. Figure 1 is a block diagram of the I/O connected to the PDP 11/23.

1.2 Feasibility of Using Previously Written Software for the Communication Link

The PDP 11/23 is one of two computers that may be used to interface the network of simulation computers to the missile seeker. The other computer, a PDP 11/34, has been used in conjunction with an SM1 seeker. It was determined that time and cost would be saved if two computers were available for missile seeker to simulation network communications. The reasoning was that while one computer is being used in testing missiles, the other computer can be readied for future testing. This arrangement would reduce preparation time needed to make a different missile's seeker available for simulation tests.



A. 16 Bit Digital output from 11/23 to missile.

B. 16 Bit Digital input to 11/23 from missile.

C. 5 Analog Signals output from 11/23 to missile.

D. 6 Analog Signals inputs to 11/23 from missile.

E. 16 Bit Digital input plus 2 control Bits output from the PDP 11/23 to a I/O control computer a PDP 11/45.

F. 16 Bit Digital output plus 2 control Bits input to the PDP 11/23 from the PDP 11/45.

G. RS-232 Input/Output

FIGURE 1 INPUT/OUTPUT CONNECTIONS TO THE PDP 11/23

It was also determined that the software previously written for the PDP 11/34 would work on the PDP 11/23 with several minor changes. The PDP 11/34 software was not fully documented, thus a considerable amount of time was spent in determining how it worked, and if it would be possible to transfer the program to run on the PDP 11/23. Appendix A contains a copy of the key program that was documented. After documenting this program, it was determined that simulations capabilities could be enhanced if new software was written. The most important enhancement was to cut the seeker computer (the 11/34 or 11/23) frame rate from 4 milliseconds to below 2 milliseconds. The result was a more powerful and flexible software package for the PDP 11/23.

1.3 Creation of a More Efficient Software Package

The new seeker computer software package has many advantages over the previous 11/34 software package. Some of these advantages are:

a. Frame rate is four times faster. The program may execute at a frame rate below 1 millisecond which is within the 2 millisecond requirement.

b. Ability to select the number of D/A and A/D channels by program control.

c. Ability to switch remote hardware in and out by computer control.

d. Light emitting diodes (LEDs) are used to show that the program is operating properly. The LEDs may also be used to enhance system debugging.

The new program is contained in Appendix B.

1.4 Program and Network Debugging

Extensive testing of the program was performed to assure that all signal levels and digital I/O could be controlled properly. To preclude damage to the missile seeker, almost all debugging and testing was conducted before the seeker was connected to the seeker computer. After open-loop tests were completed, closed loop tests were performed. At this stage, software and hardware connected to the PDP 11/23 performed as predicted. The seeker computer in conjunction with the simulation network took full control of the missile seeker.

1.5 Creation and Implementation of an Automatic Boot

The final task to which support was provided was to streamline procedures necessary to prepare the simulator seeker computer for a simulation

run. Previously it required as much as two hours for several people to prepare the seeker computer for a simulation run. It is now possible to complete this same preparation in one to two minutes. A program that permits the automatic booting of the PDP 11/23 from a single user PDP 11 computer has been the major contributing factor. The user needs only to turn on the computers and type a few control words. A program in the host PDP computer will then automatically boot the PDP 11/23 and prepare it for a simulation run.

Programs used to perform this automatic booting are contained in Appendices C, D, and E. These programs are used in conjunction with H23 the data collection and communication routine contained in Appendix B.

Appendix F contains a program named TASM. This program will be used in the near future to help reduce the time required to develop and debug programs for the seeker computer.

A Z-80 microprocessor based assembly language program named TUDUM2.MAC is contained in Appendix G. This program was used to permit a user to select from a terminal a computer-to-computer serial link. This enables the operator to command the "A" processor to boot the PDP 11/23 via a serial terminal line.

1.6 Conversion of the New PDP 11/23 Software to Run on the SM1 Seeker Computer

Research has begun for the purpose of upgrading SM1 simulations to the level of sophistication of the Harpoon simulations. A minimum amount of change to the software used for the Harpoon simulation is required. The SM1 seeker computer should be back in operation by early May, 1982.

2.0 TESTING OF CHANGES IN ERP EFFECTS ON TARGET POSITION

It was determined that small changes in the Effective Radiating Power (ERP) had negligible effect on the target position within a quad. This result was determined by adapting a FORTRAN program so that it calculates individual antenna radiating power and phase if given a target position. The program changes the ERP for this position from 30 dB to -20 dB and calculates the resulting change in target position. Appendix H contains a copy of the FORTRAN program and Appendix I contains a sample of the results.

3.0 FEASIBILITY OF ON-LINE COLOR GRAPHICS PROCESSOR

Research was conducted into the possibility of using a color graphics display system to make simulation runs more informative. An extensive search was conducted to find a system that cost less than $25,000 that would be able to display a real-time scenario of a missile flight along with target positions. Several companies such as Tektronics, Lexidata, Chromatics, and others appear to have graphic displays that will meet our real-time animation requirements.

A color graphics display system will enable users of the Central Target Simulator (CTS) to interpret results of experimentation faster and with a better perspective. At present, several slow displays are being used to show simulation results. Upgrading these display systems will make CTS more flexible and more productive for prospective users.

APPENDIX A

DLH23

0123456789012345678 ** RSX-11M V03 ** 20-APR-82 14:53:01 DR0:[212,10]DLH23.LST;7 0123456789012345678
0123456789012345678 ** RSX-11M V03 ** 20-APR-82 14:53:01 DR0:[212,10]DLH23.LST;7 0123456789012345678

(A)

```
DDDDDDDD      LL            HH      HH     222222       333333
DDDDDDDDD     LL            HH      HH     222222       333333
DD      DD    LL            HH      HH   22      22   33      33
DD      DD    LL            HH      HH   22      22   33      33
DD      DD    LL            HH      HH           22           33
DD      DD    LL            HH      HH           22           33
DD      DD    LL            HHHHHHHHHH         22           33
DD      DD    LL            HHHHHHHHHH         22           33
DD      DD    LL            HH      HH       22             33
DD      DD    LL            HH      HH       22             33
DD      DD    LL            HH      HH     22         33     33      ....
DD      DD    LL            HH      HH     22         33     33      ....
DDDDDDDDD     LLLLLLLLLLL   HH      HH   2222222222   333333       ....
DDDDDDDD      LLLLLLLLLLL   HH      HH   2222222222   333333       ....
```

A-2

```
LL              SSSSSSSS   TTTTTTTTTT       ;;;;     77777777
LL              SSSSSSSS   TTTTTTTTTT       ;;;;     77777777
LL            SS               TT           ;;;;           77
LL            SS               TT           ;;;;           77
LL            SS               TT                          77
LL            SS               TT                          77
LL              SSSSSS         TT           ;;;;         77
LL              SSSSSS         TT           ;;;;         77
LL                    SS       TT           ;;;;       77
LL                    SS       TT           ;;;;       77
LL                    SS       TT             ;;     77
LL                    SS       TT             ;;     77
LLLLLLLLLLL   SSSSSSSS         TT           ;;     77
LLLLLLLLLLL   SSSSSSSS         TT           ;;     77
```


0123456789012345678 ** RSX-11M V03 ** 20-APR-82 14:53:01 DR0:[212,10]DLH23.LST;7 0123456789012345678
0123456789012345678 ** RSX-11M V03 ** 20-APR-82 14:53:01 DR0:[212,10]DLH23.LST;7 0123456789012345678

```
                                        ;
                                        ;
                                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                        ;
                                        ;AUTHOR:        RUSSEL ANDERSON
                                        ;DATE:          SUMMER OF 1981
                                        ;
                                        ;CHANGES BY:    ROBERT B. NORMOYLE
                                        ;DATE:          21-OCT-81
                                        ;
                                        ;
                                        ;PURPOSE:       THIS PROGRAM DOWNLOADS AND CONTAINS THE HARPOON SEEKER
                                        ;       INTERFACE PROGRAM. IT IS DESIGNED TO OPERATE WITH THE HARPOON SEEKER
                                        ;       MISSILE MODEL (FAAC?).
                                        ;
                                        ;       THE HARPOON SEEKER INTERFACE PROGRAM RUNS IN THE LSI 11/23 COMPUTER
                                        ;       AND TRANSFERS DATA VIA THE PARALLEL INTERFACE ON THE 11/23 CONNECTED
                                        ;       TO THE PARALLEL INTERFACE ON THE "A" PROCESSOR.
                                        ;       THE DATA THAT IS TRANSFERED IS FOR DIGITAL AND ANALOG I/O TO THE
                                        ;       HARPOON AND ASSOCIATED PERIPHERALS.
                                        ;
                                        ;
                                        ;               NOTE:   THIS PROGRAM HAS BEEN PHASED OUT.
                                        ;
                                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                        ;
                                        ;
                                        ;
                                        ;
                                                .TITLE  DLH23
                                                .MCALL  EXIT$S,DIR$,WTSE$S,QIO$
000000  000137  006000                  ENTER:  JMP     @#6000
000004                                  PGM:    .BLKW   4000
000000                                          .ASECT
        001240                                  .=1240
001240  000000                          ST:     0
                                        ;       THIS SUBROUTINE TRANSFERS ANALOG AND DIGITAL DATA FOR OUTPUT
                                        ;       AND INPUT TO AND FROM THE HARPOON MISSILE.
001242  010067  000460                  ZRMSG:  MOV     R0,REG0         ;REG0=R0
001246  005067  000426                          CLR     FLG2PS          ;FLG2PS=0
001252  105737  167770                  ZW3:    TSTB    @#167770                ;IF DRCSR BIT 15 IS HIGH THEN
001256  100375                                  BPL     ZW3             ;CONTINUE, ELSE LOOP.
001260  013760  167774  001702                  MOV     @#167774,ZINBUF(R0)     ;ZINBUF+R0=DRINBUF
001266  005767  000406                          TST     FLG2PS          ;IF FLG2PS IS NOT SET
001272  001005                                  BNE     ZBYPAS          ;THEN BRANCH TO ZBYPAS
001274  005267  000400                          INC     FLG2PS          ;SET SECOND PASS FLAG
001300  013767  167774  000422                  MOV     @#167774,NWDS   ;NWDS=DRINBUF
001306  052737  000002  167770          ZBYPAS: BIS     #2,@#167770     ;SET CSR1 BIT 2. INTERRUPT "A" PROCESSOR.
001314  062700  000002                          ADD     #2,R0           ;R0=R0+2. ZINBUF+R0 = NEXT BUFFER LOC
001320  105737  167770                  ZW42:   TSTB    @#167770        ;WAIT UNTIL REQUEST B IS
001324  100775                                  BMI     ZW42            ;CLEARED BY "A" PROCESSOR
001326  042737  000002  167770                  BIC     #2,@#167770     ;DRINBUF(BIT 2) = 0
001334  005367  000370                          DEC     NWDS            ;NWDS=NWDS-1
001340  005767  000364                          TST     NWDS            ;IF NWDS.NE.0
001344  001342                                  BNE     ZW3             ;THEN BRANCH TO ZW3
                                        ;
                                        ;DATA INPUT FROM "A" PROCESSOR COMPLETED
```

```
                                        ;
                                        ;PREPARE FOR A/D INPUT
                                        ;
                                        ;
001346  016700  000354                  MOV     REG0,R0         ;THIS INSTRUCTION HAS NO PURPOSE!!!!!
001352  012700  004000                  MOV     #4000,R0        ;R0=A/D BUFFER
001356  005067  000316                  CLR     FLG2PS          ;FLG2PS=0. CLEAR SECOND PASS FLAG.
001362  005077  000200                  CLR     @STREG          ;CLEAR A/D CSR
001366  106427  000000                  MTPS    #0              ;PSW=0 . CLEAR PROCESSOR STATUS WORD.
001372  012767  000100  000170          MOV     #100,CH1        ;A/D CSR=#100. ENABLE DONE INTERRUPT.
001400  012737  001564  000400  CLKTST: MOV     #RET,@#400      ;PUT INTERRUPT SERVICE ROUTINE ADDRESS IN VECTR
001406  016703  000156          AG:     MOV     CH1,R3          ;MOVE A/D STATUS INFORMATION TO R3
001412  010377  000150                  MOV     R3,@STREG       ;A/D CSR=STATUS INFORMATION:CHAN #,INT ENABLED
001416  012704  000010                  MOV     #10,R4          ;WHAMOO!!! THINK ABOUT THIS!!!
001422  005277  000140                  INC     @STREG          ;A/D CSR= CSR+1. START A/D CONVERSION.
001426  000001                          *A11            ;NOTHING BETTER TO DO???
001430  105777  000132          LOOP:   TSTB    @STREG  ;I CANT BELIEVE THAT YOU ARE REALLY FINISHED.
001434  100375                          BPL     LOOP            ;IF A/D CSR(BIT 7)=1 , THEN A/D DONE
                                        ;
                                        ;A/D COMPLETED. STORE DATA IN DATA BLOCK POINTED TO BY R0
                                        ;R0=4000
                                        ;
                                        ;
001436  013720  170402                  MOV     @#170402,(R0)+          ; (R0)= A/D DATA BUFFER REGISTER (DBR)
001442  062767  000400  000120          ADD     #400,CH1                ;A/D CHANNEL # = A/D CHANNEL # + 1
001450  022700  004020                  CMP     #4020,R0                ;IF LESS THAN 16 CHANNNELS
001454  001351                          BNE     CLKTST                  ;THEN CONVERT NEXT CHANNEL
                                                                        ;ELSE CONTINUE.
                                        ;
                                        ;A/D COMPLETED. START D/A CONVERSIONS AND DIGITAL I/O.
                                        ;
                                        ;
001456  016737  000222  170440  OUTDA:  MOV     ZINBUF+2,@#170440       ;DAC 0 =ZINBUF + 2
001464  016737  000216  170442          MOV     ZINBUF+4,@#170442       ;DAC1=ZINBUF+4
001472  016737  000212  170444          MOV     ZINBUF+6,@#170444       ;DAC2=ZINBUF+6
001500  016737  000206  170446          MOV     ZINBUF+10,@#170446      ;DAC3=ZINBUF+10
                                        ;
                                        ;               BEGIN DIGITAL I/O
                                        ;
001506  016737  000204  167762          MOV     ZINBUF+14,@#167762      ;DRV11 #2 DROUTBUF=ZINBUF+14
001514  013737  167764  004020          MOV     @#167764,@#4020         ;(4020)=DRV11 #2  DRINBUF
001522  032737  000020  004020          BIT     #20,@#4020              ;ZERO FLAG=1 IF (4020).AND.#20 = #20
001530  001004                          BNE     CLBIT                   ;PC=CLBIT IF ZERO FLAG = 0
001532  052737  000020  004020          BIS     #20,@#4020              ;(4020)BIT 4 = 1
001540  000403                          BR      SBIT                    ;PC=SBIT
001542  042737  000020  004020  CLBIT:  BIC     #20,@#4020              ;(4020)BIT 4 = 0
001550  005137  004020          SBIT:   COM     @#4020                  ;(4020)= ONE'S COMPLEMENT OF (4020)
001554  005067  000010                  CLR     CH1                     ;(CH1)=0
001560  000167  000006                  JMP     SMSG                    ;PC=SMSG
                                        ;
                                        ;
                                        ;
                                        ;                       DATA STORAGE
                                        ;
                                        ;
001564  000002          RET:    RTI                                     ;PC=(SP)+    PS=(SP)+
001566  170400          STREG:  170400                                  ;ADV11-A CSR IS AT LOCATION 170400
```

```
001570  000000                  CH1:    0                                   ;CHANNEL INFORMATION
                                ;
                                ;
                                ;                       SEND DATA TO "A" PROCESSOR
                                ;
001572  010067  000130          SMSG::  MOV     R0,REG0         ;(REG0)=R0.  SAVE R0. WHY?
001576  005000                          CLR     R0              ;R0=0           WHY?
001600  012767  000012  000122          MOV     #10.,NWDS       ;NWDS=NUMBER OF WORDS TO BE TRANSFERRED=10.
001606  012700  003776                  MOV     #4000-2,R0      ;R0=#4000-2 = ONE WORD BEFORE DATA BLOCK
001612  012710  000012                  MOV     #10.,(R0)       ;(R0)=NUMBER OF WORDS TO BE TRANSFERRED=10.
                                ;                       TRANSFER DATA VIA DRV11 #1
001616  012037  167772          NXTWD:  MOV     (R0)+,@#167772  ;DRV11 #1 DROUTBUF=(R0)
001622  052737  000001  167770          BIS     #1,@#167770     ;DRV11 #1 DRCSR=1 TELL "A" PROC NEW WORD READY
001630  005737  167770          W4:     TST     @#167770                ;IF "A" PROCESSOR RECEIVED WORD THEN
001634  100375                          BPL     W4              ;CONTINUE, ELSE WAIT UNTIL WORD RECEIVED.
001636  042737  000001  167770          BIC     #1,@#167770     ;DRV11 #1 BIT 1 =0 . CSR0=0
001644  005737  167770          W5:     TST     @#167770        ;IF BIT 15 OF DRSCR = 0 (SET BY "A" PROC)
001650  100775                          BMI     W5              ;THEN CONTINUE, ELSE WAIT UNTIL RESET
                                ;
001652  005367  000052                  DEC     NWDS            ;NWDS=NWDS-2. DECREMENT THE NUMBER OF WORDS
001656  005767  000046                  TST     NWDS            ;IF NWDS=0 THEN
001662  001355                          BNE     NXTWD           ;CONTINUE, ELSE TRANSMIT NEXT WORD TO "A" PROC
001664  016700  000036                  MOV     REG0,R0 ;R0=REG0 . RESTORE CONTENTS OF R0. WASTE SOME MORE TIME.
001670  000167  177346                  JMP     ZRMSG   ;PC=ZRMSG. CLOSED READ WRITE LOOP.
                                ;END SEEKER COMPUTER CODE.
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;                       BEGIN "A" PROCESSOR DATA
                                ;
                                ;
001674  000472                  ZNWDS:  ENLD-ST
001676  000000                  ZREG0:  0
001700  000000                  FLG2PS: 0
001702                          ZINBUF: .BLKW   10.
001726  000000                  REG0:   0
001730  000000                  NWDS:   0
001732  000000                  ENLD:   0
                                ;
                                ;                       "A" PROCESSOR PROGRAM THAT TRANSFERS
                                ;                       LSI 11/23 PROGRAM TO SEEKER COMPUTER
                                ;
                                ;
001734                                  .ASECT
        167660                          .=167660
167660  000000                  HCSR1:  0
167662  000000                  OUTBUF: 0
```

```
                006000                          .=6000
006000  006267  173670          START:  ASR     ZNWDS           ;ZNWDS=ZNWDS/2
006004  016767  173664  173226          MOV     ZNWDS,ST        ;ST=ZNWDS
006012  010067  173660          ZSMSG:  MOV     R0,ZREG0        ;ZREG0=R0
006016  005000                          CLR     R0              ;R0=0
006020  016067  001240  161634  ZNXTWD: MOV     1240(R0),OUTBUF ;OUTBUF=(1240)+R0
006026  052767  000001  161624          BIS     #1,MCSR1                ;INITIATE DATA TRANSFER
006034  005767  161620          ZW4:    TST     MCSR1           ;WAIT UNTIL SEEKER COMPUTER HAS
006040  100375                          BPL     ZW4                     ;RECEIVED DATA
006042  042767  000001  161610          BIC     #1,MCSR1                ;CLEAR NEW DATA AVAILABLE BIT
006050  005767  161604          ZW5:    TST     MCSR1           ;WAIT UNTIL SEEKER COMPUTER HAS
006054  100775                          BMI     ZW5             ;ACKNOWLEDGED THAT THERE IS NO NEW DATA
006056  062700  000002                  ADD     #2,R0           ;R0=R0+2
006062  005367  173606                  DEC     ZNWDS           ;ZNWDS=ZNWDS-1
006066  005767  173602                  TST     ZNWDS           ;IF ZNWDS=0
006072  001352                          BNE     ZNXTWD          ;THEN CONTINUE, ELSE TRANSFER NEXT WORD
006074  016700  173576                  MOV     ZREG0,R0                ;R0=ZREG0 . RESTORE R0
006100                                  EXIT$S
000000'                                 .END    ENTER
```

```
DLH23   MACRO M11    20-APR-82 14:52  PAGE 1-4
SYMBOL TABLE

AG      001406          LOOP    001430          MEGO    001726          M4      001630          ZRMSG   001242
CH1     001570          MCSR1   167660          MET     001564          M5      001644          ZSMSG   006012
CLBIT   001542          MWDS    001730          SBIT    001550          ZMYPAS  001306          Zw3     001252
CLMTST  001400          NXTWD   001616          SMSG    001572 G        ZINBUF  001702          Zw4     006034
ENLD    001732          OUTBUF  167662          SI      001240          ZNWDS   001674          Zw42    001320
ENTER   000000R         OUTDA   001456          START   006000          ZNXTWD  006020          Zw5     006050
FLG2PS  001700          PGM     000004R         STMEG   001566          ZREGO   001676

. ABS.  167664      000
        010004      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  544 WORDS  ( 3 PAGES)
DYNAMIC MEMORY:  3162 WORDS  ( 12 PAGES)
ELAPSED TIME:  00:00:04
DLH23,DLH23=DLH23
```

APPENDIX B

H23

B.

```
HH      HH     222222       333333
HH      HH     222222       333333
HH      HH   22      22   33      33
HH      HH   22      22   33      33
HH      HH           22           33
HH      HH           22           33
HHHHHHHHHH         22           33
HHHHHHHHHH         22           33
HH      HH       22               33
HH      HH       22               33
HH      HH     22           33      33      ....
HH      HH     22           33      33      ....
HH      HH   2222222222     333333          ....
HH      HH   2222222222     333333          ....
```

B-2

```
LL             SSSSSSSS   TTTTTTTTTT       ;;;;         11       5555555555
LL             SSSSSSSS   TTTTTTTTTT       ;;;;         11       5555555555
LL           SS               TT          ;;;;       1111       55
LL           SS               TT          ;;;;       1111       55
LL           SS               TT                       11       555555
LL           SS               TT                       11       555555
LL             SSSSSS         TT          ;;;;         11            55
LL             SSSSSS         TT          ;;;;         11            55
LL                   SS       TT          ;;;;         11              55
LL                   SS       TT          ;;;;         11              55
LL                   SS       TT            ;;         11       55     55
LL                   SS       TT            ;;         11       55     55
LLLLLLLLLL   SSSSSSSS         TT          ;;         111111       555555
LLLLLLLLLL   SSSSSSSS         TT          ;;         111111       555555
```

```
                                .MACRO  PUSH    REG1,REG2,REG3,REG4,REG5,REG6,REG7,REG8
                                .NARG   MPUSH
                                .IF     NE,MPUSH
                        ;
                        ;
                        ;
                                MOV     REG1,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     REG2,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     REG3,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     REG4,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     REG5,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     REG6,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     REG7,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     REG8,-(SP)
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                        ;
                        ;
                        ;
                                .ENDM
                        ;
                        ;
                        ;
                                .MACRO  POP     REG0,REG1,REG2,REG3,REG4,REG5,REG6,REG7
                        ;
                        ;
                        ;
                                .NARG   MPOP
                                .IF     NE,MPOP
                                MOV     (SP)+,REG0
                                .IF     NE,MPOP-1
                                MOV     (SP)+,REG1
                                .IF     NE,MPOP-2
                                MOV     (SP)+,REG2
                                .IF     NE,MPOP-3
                                MOV     (SP)+,REG3
```

```
                                .IF     NE,MPUP-4
                                MOV     (SP)+,REG4
                                .IF     NE,MPUP-5
                                MOV     (SP)+,REG5
                                .IF     NE,MPUP-6
                                MOV     (SP)+,REG6
                                .IF     NE,MPUP-7
                                MOV     (SP)+,REG7
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                        ;
                                .ENDM
177776                  PSW=177776              ;PROCESSOR STATUS WORD
167770                  DRVCSR=167770                   ;DRV #1 CONTROL STATUS REGISTER
167772                  DRVOUT=167772                   ;DRV #1 16-BIT OUTPUT CHANNEL
167774                  DRVIN=167774                    ;DRV #1 16-BIT INPUT CHANNEL
                        ;
                        ;               DRV #2-USED FOR DIGITAL I/O WITH SEEKER
167760                  DIGCSR=167760                   ;DRV #2 CONTROL STATUS REGISTER
167762                  DIGOUT=167762                   ;DRV #2 16-BIT OUTPUT CHANNEL
167764                  DIGIN=167764                    ;DRV #2 16-BIT INPUT CHANNEL
                        ;
                        ;               AAV11-A DIGITAL ANALOG CARD. OUTPUT CONNECTED TO SEEKER
170440                  DA0=170440                      ;DIGITAL/ANALOG CHANNEL 0
                        ;       :
                        ;       :
170456                  DA7=170456                      ;LAST DIGITAL TO ANALOG CHANNEL: CHANNEL 7
                        ;
                        ;               ADV11-A ANALOG/DIGITAL CARD. INPUT FROM SEEKER
004000                  ADBUF=4000                      ;ADBUF(0) TO ADBUF(20 OCT)=CONV DATA STORAGE
170400                  ADCSR=170400                    ;ADV11 CONTROL STATUS REGISTER
170402                  ADVDBR=170402                   ;ADV11 CONVERTED DATA BUFFER
000000                  CHAN0=0                         ;CHAN0=CODE TO SELECT A/D CON ON CHAN 0
000400                  CHAN1=400                       ;CHAN1=CODE TO SELECT A/D CON ON CHAN 1
                        ;
                        ;
                        ;                    CSR1   :   CSR0
000000                  C00=0           ;       0   :   0
000001                  C01=1           ;       0   :   1
000002                  C10=2           ;       1   :   0
000200                  CA=200          ;REQ A SET
100000                  CB=100000       ;REQ B SET
000201                  CA01=201        ;REQ A AND CSR0 SET
100001                  CB01=100001     ;REQ B AND CSR0 SET
100002                  CB10=100002     ;REQ B AND CSR1 SET
100202                  CBA10=100202    ;REQ A AND REQ B SET AND CSR0 AND CSR1 SET
                        ;
000001                  ONCSR0=1
000002                  ONCSR1=2
                        ;
                        ;
```

```
                                ;
                                ;
                                ;       SEEKER INITIALIZATION CONDITIONS
                                ;
177047          AVOFF=177047                    ;AVIONICS POWER OFF.
177247          RADOFF=177247                   ;RADIATE OFF, AVIONICS POWER ON
007720          FRANGE=7720                     ;FULL RANGE=
004000          MINR0=4000                      ;MIN RANGE EXTENT =     V
004000          AZ0=4000                        ;AZIMUTH =    DEG
003365          ELM6=3365                       ;ELEVATION =  DEG
004674          AZE10=4674                      ;AZIMUTH EXTENT=     DEG
005570          DEG20 =5570                     ;DEG20=      DEGREES
000131          DEG2=131                        ;DEG2= DEG DELTA
                                ;
                                ;
```

```
                                        .TITLE  M23
                                        .GLOBL  TCOUNT
                        ;
000000  001016          TCOUNT: EXIT-TCOUNT
                        ;
                        ;
                        ;
000002                  M23:            ;SEEKER AND 11/23 INITIALIZATION
                        ;
                        ;
                        ;
000002  012706  020000          MOV     #20000,SP       ;STACK POINTER = 20000 OCTAL
                        ;
                        ;
                        ;               INITIALIZE SEEKER
000006  012737  177047  167762          MOV     #AVOFF,@#DIGOUT         ;TURN AVIONICS POWER OFF
000014  012737  004000  170440          MOV     #AZ0,@#DAU              ;AZIMUTH=0 DEG
000022  012737  003365  170442          MOV     #ELM6,@#DAU+2           ;ELEVATION = -6 DEG
000030  012737  004000  170444          MOV     #MINR0,@#DAU+4          ;MINIMUM RANGE EXTENT = 0
000036  012737  004674  170450          MOV     #AZE10,@#DAU+10         ;AZIMUTH EXTENT = +/- 10 DEG
000044  012737  007720  170452          MOV     #FRANGE,@#DAU+12        ;RANGE= FULL=+10.0VDC
                        ;
000052  005037  167770  BEGLP:  CLR     @#DRVCSR
000056  012737  000001  170456  TEST1:  MOV     #1,@#DAU+16             ;TEST1 FOR TIMING LOOP W/O-SCOPE
000064  022737  000000  167770  WAIT00: CMP     #C00,@#DRVCSR           ;IF DRVCSR=0 THEN SYSTEM
000072  001403                  BEQ     TEST2
000074  004767  000570          CALL    LIGHTF          ;SHOW EM ITS WORKING
000100  000771                  BR      WAIT00          ;SET UP OKAY
                        ;
000102  005037  170456  TEST2:  CLR     @#DAU+16                ;TEST2 SAME PURPOSE AS TEST1
```

```
163                                     ;               IS 11/45 READY TO SEND DATA
164                                     ;       IF NOT COLLECT A/D DATA UNTIL READY
165                                     ;
166                                     ;
167 000106  012701  004000          ADBEG:  MOV     #ADBUF,R1       ;R1=BEG OF A/D DATA STORAGE BLOCK
168 000112  005003                  CLR     R3              ;R3=CHANNEL SELECTOR
169 000114  010337  170400          ADCOL:  MOV     R3,@#ADCSR      ;ADCSR=CHANNEL NUMBER
170 000120  005237  170400                  INC     @#ADCSR         ;START A/D CONVERSION
171 000124  062703  000400                  ADD     #400,R3         ;R3=NEXT A/D CHANNEL
172                                     ;               IS 11/45 READY TO SEND DATA
173 000130  022737  000200  167770          CMP     #CA,@#DRVCSR    ;IF REQ A SET THEN EXIT A/D LOOP
174 000136  001421                          BEQ     INIT
175 000140  022701  004016                  CMP     #ADBUF+16,R1    ;DOES R1=ADBUF+16
176 000144  001406                          BEQ     ADEND           ;IF YES, THEN COLLECT LAST
177                                                                 ;DATA. JUMP TO ADBEG
178 000146  105737  170400          ADLOOP: TSTB    @#ADCSR         ;IS A/D CONVERSION COMPLETED
179 000152  100375                          BPL     ADLOOP          ;WAIT UNTIL CONVERSION COMPLETED
180 000154  013721  170402                  MOV     @#ADVDBR,(R1)+  ;(R1) = A/D STORAGE BUFFER
181                                     ;
182 000160  000755                          BR      ADCOL           ;COLLECT NEXT CHANNEL
183                                     ;
184                                     ;               LAST A/D CHANNEL
185 000162  105737  170400          ADEND:  TSTB    @#ADCSR         ;IS A/D CONVERSION COMPLETED
186 000166  100375                          BPL     ADEND           ;WAIT UNTIL COMPLETED
187 000170  013721  170402                  MOV     @#ADVDBR,(R1)+  ;(R1)=LAST LOC IN A/D STORAGE
188                                                                 ;       BUFFER
189 000174  004767  000500                  CALL    LIGHTS
190 000200  000742                          BR      ADBEG           ;COLLECT NEXT SET OF DATA
191                                     ;
192                                     ;
```

```
                                        ;
                                        ;
                                        ;
                                        ;       REINITIALIZE DATA POINTERS
000202  012700  170440          INIT:   MOV     #DAC,R0                 ;R1=LOC OF D/A REGISTERS
000206  012701  004000                  MOV     #ADBUF,R1                       ;R1=LOC OF A/D MEMORY
000212  013703  167774                  MOV     @#DRVIN,R3              ;R3= # OF WORDS TO TRANSFER
                                        ;
000216  013737  167764  167772          MOV     @#DIGIN,@#DRVOUT        ;DRVOUT=DIGITAL I/O DATA
000224  012737  000001  167770          MOV     #UNCSR0,@#DRVCSR                ;CSR1=0,CSR0=1
000232  005303                          DEC     R3
000234  001474                          BEQ     DIGFIN
                                        ;
                                        ;
                                        ;WAIT UNTIL REQB=0 ANA REQA=1
                                        ;
000236  022737  100001  167770  WTB:    CMP     #CB01,@#DRVCSR          ;DOES DRVCSR=CB01
000244  001374                          BNE     WTB                     ;NO, GOTO WTB
                                                                        ;YES,CONTINUE
                                        ;               IS RADIATE OFF
000246  105737  167774                  TSTB    @#DRVIN                 ;IF AVPWRON(BIT 7) SET THEN CONTINUE
000252  001476                          BEQ     SHUTDN                  ;IF NOT, THEN SHUT DOWN SEEKER
                                        ;;
                                        ;
                                        ;               RECEIVE DIGITAL OUTPUT WORD
000254  013737  167774  167762          MOV     @#DRVIN,@#DIGOUT        ;DIGOUT=DRVIN
                                        ;OUTPUT FIRST ANALOG/DIGITAL DATA TO "A" PROCESSOR
000262  012137  167772                  MOV     (R1)+,@#DRVOUT          ;DRVOUT=A/D DATA CHANNEL 0
000266  012737  000002  167770          MOV     #UNCSR1,@#DRVCSR        ;CSR1=1, CSR0=0
000274  005303                          DEC     R3
000276  001433                          BEQ     FINISH
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;               READ/WRITE NEXT I/O WORD
000300  022737  100202  167770  WAITBA: CMP     #CBA10,@#DRVCSR         ;DOES DRVCSR=CBA10
000306  001374                          BNE     WAITBA                  ;NO,GOTO WAITBA
                                                                        ;YES, CONTINUE
000310  013720  167774                  MOV     @#DRVIN,(R0)+           ;(R0)=DIG/ANALOG DATA. R0=R0+2
000314  012137  167772                  MOV     (R1)+,@#DRVOUT          ;DRVOUT=ANALOG/DIG DATA. R1=R1+2
000320  012737  000001  167770          MOV     #UNCSR0,@#DRVCSR                ;CSR1=1, CSR0=0
000326  005303                          DEC     R3
000330  001426                          BEQ     FINBA
                                                                        ;ELSE
                                        ;
                                        ;               READ/WRITE NEXT I/O WORD
000332  022737  100001  167770  WAITB:  CMP     #CB01,@#DRVCSR          ;DOES DRVCSR=CB01
                                        ;
000340  001374                          BNE     WAITB           ;NO, GOTO WAITB
                                                                        ;YES, CONTINUE
                                        ;INPUT ANALOG WORD FROM "A" PROCESSOR
000342  013720  167774                  MOV     @#DRVIN,(R0)+           ;
000346  012137  167772                  MOV     (R1)+,@#DRVOUT          ;
000352  012737  000002  167770          MOV     #UNCSR1,@#DRVCSR                ;CSR1=1, CSR0=0
```

```
  251 000360  005303                        DEC     R3
  252 000362  001401                        BEQ     FINISH
  253                               ;
  254 000364  000745                        BR      WAITBA
  255                               ;
  256                               ;
  257                               ;
  258                               ;
  259                               ;
  260                               ;
  261                               ;               ALL BUT ONE CHANNEL OF DATA HAVE BEEN RECEIVED
  262 000366  022737  100202  167770 FINISH: CMP     #CBA10,@#DRVCSR          ;DOES DRVCSR=C10
  263 000374  001374                        BNE     FINISH                   ;NO, GOTO FINISH
  264                                                                        ;YES, CONTINUE
  265                               ;WRITE LAST A/D CHANNEL TO "A" PROCESSOR
  266 000376  013710  167774                MOV     @#DRVIN,(R0)
  267 000402  000167  177444                JMP     BEGLP
  268                               ;
  269 000406  022737  100001  167770 FINBA:  CMP     #CB01,@#DRVCSR
  270 000414  001374                        BNE     FINBA
  271 000416  013710  167774                MOV     @#DRVIN,(R0)
  272 000422  000167  177424                JMP     BEGLP
  273                               ;
  274                               ;
  275                               ;               JUST DESIRE DIGITAL INFORMATION
  276                               ;
  277 000426  022737  000201  167770 DIGFIN: CMP     #CA01,@#DRVCSR
  278 000434  001374                        BNE     DIGFIN
  279 000436  013737  167774  167762        MOV     @#DRVIN,@#DIGOUT
  280 000444  000167  177402                JMP     BEGLP
  281                               ;
  282                               ;
  283                               ;
  284                               ;
  285                               ;
```

```
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
000450                                  SHUTDN:         ;SHUTDOWN SEEKER COMPUTER
                                        ;               IF THE LIGHT ARRAY MONITORING DRV #1 HAS A STRING OF 2 OR 3
                                        ;               BITS MOVING VERY FAST, THEN THE COMPUTER IS STUCK IN THIS
                                        ;               ROUTINE. IF THIS ROUTINE IS WORKING PROPERLY, THEN THIS
                                        ;               LIGHT PATTERN INDICATES THAT THE ANT IS NOT POSITIONING
                                        ;               CORRECTLY TO AZ=20 DEG +/- 2 DEG, AND EL=20 DEG +/- 2 DEG
                                        ;
                                        ;                       TURN RADIATE OFF
000450  012737  177247  167762          ;       MOV     #RADOFF,@#DIGOUT                ;TURN RADIATE OFF. THIS
                                        ;                                               CAUSES ANT POS TO GOTO AZ=20, EL= 20
000456  012701  004000                  ;       MOV     #ADBUF,R1                       ;R1=DEST BUFFER FOR A/D DATA
000462  012702  000001                  ;       MOV     #1,R2                           ;R2=#1=NUMBER OF CHANNELS TO COLLECT
000466  012703  000000                  ;       MOV     #CHAN0,R3                       ;R3=FIRST CHANNEL TO CONVERT
                                        ;
                                        ;                       CHANGE LIGHT PATTERN
000472  012767  000003  000310          ;       MOV     #3,LIGHTL                       ;SHIFTING 2 BITS
000500  012767  140000  000304          ;       MOV     #140000,LIGHTR                  ;LEFT AND RIGHT
                                        ;                       WAIT UNTIL AZIMUTH SET TO 20 DEG +/- 2 DEG
000506                                  AZSET:
000506  004767  000156                  ;       CALL    LIGHTF
000512  004767  000104                  ;       CALL    ADCONV
000516  022711  005437                  ;       CMP     #DEG20-DEG2,(R1)                ;IS AZ POS > 18 DEG
000522  100371                          ;       BPL     AZSET                           ;NO, DO LOOP UNTIL SET
                                        ;
000524  022711  005721                  ;       CMP     #DEG20+DEG2,(R1)                ;IS AZ POS < 22 DEG
000530  100766                          ;       BMI     AZSET                           ;NO, LOOP UNTIL SET
                                        ;
                                        ;
                                        ;                       CHANGE LIGHT PATTERN
000532  012767  000007  000250          ;       MOV     #7,LIGHTL
000540  012767  160000  000244          ;       MOV     #160000,LIGHTR                  ;SHIFTING 3 BIT TRAIN IN LIGHTF
                                        ;
000546  012703  000400                  ;       MOV     #CHAN1,R3                       ;R3=CHAN1=SELECT CHANNEL 1
000552  012701  004002                  ;       MOV     #ADBUF+2,R1                     ;R1=ADBUF(2)=ELEVATION GIMBAL
000556                                  ELSET:          ;WAIT UNTIL ELEVATION SET TO 20 DEG +/- 2 DEG
                                        ;
                                        ;
000556  004767  000106                  ;       CALL    LIGHTF                          ;SHOW OPERATOR THAT PROGRAM IS LOOPING
000562  004767  000034                  ;       CALL    ADCONV                          ;CONVERT A/D DATA
000566  022711  005437                  ;       CMP     #DEG20-DEG2,(R1)                ;IS EL > 18 DEG
000572  100371                          ;       BPL     ELSET                           ;NO, LOOP UNTIL ELEV >18 DEG
000574  022711  005721                  ;       CMP     #DEG20+DEG2,(R1)                ;IS EL < 22 DEG
000600  100766                          ;       BMI     ELSET                           ;NO, LOOP UNTIL ELEV < 22 DEG
                                        ;
                                        ;
                                        ;                       RESET LIGHTS TO A/D LOOP PATTERN
000602  012767  000001  000200          ;       MOV     #1,LIGHTL                       ;SINGLE LIGHT SHIFTING LEFT AND RIGHT
000610  012767  100000  000174          ;       MOV     #100000,LIGHTR
```

```
344 000016  000167  177160                  JMP     M23             ;RESTART PROGRAM:(1) TURN OF AVRAM
345                                                                 ;                (2) INITIALIZE D/A
346                                                                 ;                (3) INIT DIG OUTPUT
347                                                                 ;                (4) WAIT IN A/D LOOP
348                                 ;
349                                 ;
350                                 ;
351                                 ;
```

```
                        ;
                        ;       ADCONV-USES THE ADV11-A TO CONVERT ANALOG DATA TO DIGITAL
                        ;
                        ;       ON INPUT-       R1=DESTINATION BUFFER TO PLACE CONVERTED DATA
                        ;                       R2=#1=NUMBER OF CHANNELS TO CONVERT
                        ;                       R3=FIRST CHANNEL: 0,400,1000,1400,...
                        ;                         CHANNEL NUMBER: 0,1  ,2    ,3   ,...
                        ;
                        ;
000622                  ADCONV: PUSH    R1,R2,R3
        000003                  .NARG   MPUSH
                                .IF     NE,MPUSH
                        ;
                        ;
                        ;
000622  010146                  MOV     R1,-(SP)
        000002          MPUSH=MPUSH-1
                                .IF     NE,MPUSH
000624  010246                  MOV     R2,-(SP)
        000001          MPUSH=MPUSH-1
                                .IF     NE,MPUSH
000626  010346                  MOV     R3,-(SP)
        000000          MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     ,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     ,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     ,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     ,-(SP)
                        MPUSH=MPUSH-1
                                .IF     NE,MPUSH
                                MOV     ,-(SP)
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                                .ENDC
                        ;
                        ;
                        ;
000630  010337  170400          MOV     R3,@#ADCSR              ;ADCSR=FIRST CHANNEL
000634  005237  170400  ADCNV1: INC     @#ADCSR                 ;INITIALIZE CONVERSION
000640  062703  000400          ADD     #CHAN1,R3                       ;R3=NEXT CHANNEL
000644  105737  170400  ADCNV2: TSTB    @#ADCSR                 ;WAIT UNTIL CONV COMPLETED
000650  100375                  BPL     ADCNV2
                        ;
000652  013721  170402          MOV     @#ALVDBF,(R1)+          ;SAVE CONVERTED DATA
000656  077212                  SOB     R2,ADCNV1               ;DO NEXT CHANNEL?
000660                          POP     R3,R2,R1                ;RTN
```

```
                                        ;
                                        ;
                                        ;
                000003                          .NARG   MPUP
                                                .IF     NE,MPUP
000660  012603                                  MOV     (SP)+,R3
                                                .IF     NE,MPUP-1
000662  012602                                  MOV     (SP)+,R2
                                                .IF     NE,MPUP-2
000664  012601                                  MOV     (SP)+,R1
                                                .IF     NE,MPUP-3
                                                MOV     (SP)+,
                                                .IF     NE,MPUP-4
                                                MOV     (SP)+,
                                                .IF     NE,MPUP-5
                                                MOV     (SP)+,
                                                .IF     NE,MPUP-6
                                                MOV     (SP)+,
                                                .IF     NE,MPUP-7
                                                MOV     (SP)+,
                                                .ENDC
                                                .ENDC
                                                .ENDC
                                                .ENDC
                                                .ENDC
                                                .ENDC
                                                .ENDC
                                                .ENDC
                                        ;
  372 000666  000207                            RETURN                          ;RETURN
  373                                   ;
  374                                   ;;;;;
  375                                   ;
```

```
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;       SPECIAL EFFECTS
                                        ;       LIGHTS  IS A PROGRAM USED TO SHOW OPERATOR THAT
                                        ;       THE LSI 11/23 PROGRAM IS RUNNING PROPERLY
                                        ;
000670  012767  000050  000110  LIGHTF: MOV     #50,SPEED
000676  000403                         BR      LIT1
000700                          LIGHTS:
000700  012767  000330  000100          MOV     #330,SPEED
000706                          LIT1:
000706  005367  000102                  DEC     COUNT
000712  001401                          BEQ     LIT2
000714  000207                          RETURN
000716                          LIT2:
000716  000241                          CLC                     ;CLEAR CARRY FLAG
000720  006167  000064                  ROL     LIGHTL          ;ROTATE LIGHTL LEFT
000724  103003                          BCC     NCL
000726  052767  000001  000054          BIS     #1,LIGHTL
000734                          NCL:
000734  016767  000050  000042          MOV     LIGHTL,LSTOR    ;SET BIT ON DRVOUT LED'S
000742  000241                          CLC                     ;CLEAR CARRY FLAG
000744  006067  000042                  ROR     LIGHTR          ;ROTATE LIGHTR RIGHT
000750  103003                          BCC     NCR
000752  052767  100000  000032          BIS     #100000,LIGHTR
000760                          NCR:
000760  056767  000026  000016          BIS     LIGHTR,LSTOR    ;OR LIGHTR WITH @#DRVOUT
000766  016737  000012  167772          MOV     LSTOR,@#DRVOUT  ;OUTPUT NEW LIGHTS
000774  016767  000006  000012          MOV     SPEED,COUNT
001002  000207                          RETURN                  ;RETURN
                                        ;
001004  000000                  LSTOR:  .WORD   0
                                        ;
001006  000000                  SPEED:  .WORD   0
001010  000001                  LIGHTL: .WORD   1
001012  100000                  LIGHTR: .WORD   100000
001014  001000                  COUNT:  .WORD   1000
001016  001016                  EXIT:   .WORD   EXIT-COUNT
        000001                          .END
```

```
M23     MACRO M11    20-APR-82 15:19  PAGE 7-1
SYMBOL TABLE

ADBEG   000106R        CA     = 000200        DEG2   = 000131        FRANGE= 007720        NCR     000760R
ADBUF = 004000          CA01   = 000201        DEG20  = 005570        M23     000002R       UNCSR0= 000001
ADCNV1  000634R         CB     = 100000        DIGCSR= 167760         INIT    000202R       UNCSR1= 000002
ADCNV2  000644R         CBA10  = 100202        DIGFIN  000426R        LIGHTF  000670R       PSW   = 177776
ADCOL   000114R         CB01   = 100001        DIGIN  = 167764        LIGHTL  001010R       RADOFF= 177247
ADCONV  000622R         CB10   = 100002        DIGOUT= 167762         LIGHTR  001012R       SHUTDN  000450R
ADCSR = 170400          CHAN0  = 000000        DRVCSR= 167770         LIGHTS  000700R       SPEED   001006R
ADEND   000162R         CHAN1  = 000400        DRVIN  = 167774        LIT1    000706R       TCOUNT  000000RG
ADLOOP  000146R         COUNT    001014R       DRVOUT= 167772         LIT2    000716R       TEST1   000056R
ADVDBR= 170402          C00    = 000000        ELMO   = 003365        LSTOR   001004R       TEST2   000102R
AVOFF = 177047          C01    = 000001        ELSET   000556R        MINRO = 004000        WAITB   000332R
AZERO = 004674          C10    = 000002        EXIT    001016R        MPOP  = 000003        WAITBA  000300R
AZSET   000506R         DA0    = 170440        FINBA   000406R        MPUSH = 000000        WAITOO  000064R
AZO   = 004000          DA7    = 170456        FINISH  000366R        NCL     000734R       WTB     000236R
BEGLP   000052R

. ABS.  000000      000
        001020      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  952 WORDS  ( 4 PAGES)
DYNAMIC MEMORY:  3162 WORDS  ( 12 PAGES)
ELAPSED TIME:  00:00:06
M23,M23=M23
```

APPENDIX C
BOOT

```
BBBBBBBB      000000      000000     TTTTTTTTTT
BBBBBBBB      000000      000000     TTTTTTTTTT
BB      BB    00    00    00    00       TT
BB      BB    00    00    00    00       TT
BB      BB    00    00    00    00       TT
BB      BB    00    00    00    00       TT
BBBBBBBB      00    00    00    00       TT
BBBBBBBB      00    00    00    00       TT
BB      BB    00    00    00    00       TT
BB      BB    00    00    00    00       TT
BB      BB    00    00    00    00       TT          ....
BB      BB    00    00    00    00       TT          ....
BBBBBBBB      000000      000000         TT          ....
BBBBBBBB      000000      000000         TT          ....
```



```
LL            SSSSSSSS   TTTTTTTTTT      ;;;;      333333
LL            SSSSSSSS   TTTTTTTTTT      ;;;;      333333
LL          SS             TT           ;;;;    33      33
LL          SS             TT           ;;;;    33      33
LL          SS             TT                           33
LL          SS             TT                           33
LL            SSSSSS       TT           ;;;;          33
LL            SSSSSS       TT           ;;;;          33
LL                  SS     TT           ;;;;            33
LL                  SS     TT           ;;;;            33
LL                  SS     TT             ;;    33      33
LL                  SS     TT             ;;    33      33
LLLLLLLLLL  SSSSSSSS       TT           ;;        333333
LLLLLLLLLL  SSSSSSSS       TT           ;;        333333
```

```
                                        .TITLE  BOOT
                                        .GLOBL  SBOOT,DLOAD,PSEND,CINIT
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;
                                ;AUTHOR:        ROBERT B. NORMOYLE
                                ;               QUEST RESEARCH CORP
                                ;               MCLEAN,VA
                                ;               PHONE: 821-3200
                                ;
                                ;DATE:          APRIL 1982
                                ;
                                ;PURPOSE:       THIS ROUTINE IS THE MAIN ROUTINE WHICH CALLS
                                ;               SUBROUTINES TO AUTOMATICALLY BOOT THE 11/23
                                ;               MICROCOMPUTER. IT FIRST CALLS SBOOT WHICH PERFORMS
                                ;               A SERIAL BOOT VIA THE TERMINAL LINE.  AFTER THE
                                ;               SERIAL BOOT IS COMPLETE THE 11/23 IS CAPABLE OF
                                ;               RUNNING A PROGRAM TO PERFORM A PARALEL BOOT. THIS
                                ;               ROUTINE THEN CALL DLOAD WHICH DOWNLOADS THE PROGRAM
                                ;               H23 VIA THE PARALLEL LINK. WHEN THIS IS COMPLETED
                                ;               THIS PROGRAM SENDS A MESSAGE TO THE CONSOLE
                                ;               TERMINAL TO NOTIFY THAT BOOT WAS SUCCESSFUL.
                                ;
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;
000000  000002'                 FIN:    .WORD   FIN+2
000002     040     040     040  FIN2:   .ASCIZ  /     BOOT COMPLETED/<BELL><CR><LF><LF>
000005     040     040     102
000010     117     117     124
000013     040     103     117
000016     115     120     114
000021     105     124     105
000024     104     007     015
000027     012     012     000
                                        .EVEN
000032  177777                          .WORD   177777
        000015                  CR=15
        000012                  LF=12
        000007                  BELL=07
000034  004767  000000G         BOOT:   CALL    SBOOT           ;SERIAL BOOT THE LSI
000040  004767  000000G                 CALL    DLOAD
000044  004767  000000G                 CALL    CINIT           ;PREPARE COMMUNICATIONS WITH CONSOLE
000050  012737  000100  177560          MOV     #100,@#177560
000056  016701  177716                  MOV     FIN,R1          ;PRINT "BOOT COMPLETED"
000062  004767  000000G                 CALL    PSEND
                                ;
000066  012746  000463                  MOV     #463,-(SP)      ;RETURN TO SYSTEM OPERATION
000072  104377                          EMT     377
        000034'                         .END    BOOT
```

```
BOOT    MACRO M11    20-APR-82 15:29  PAGE 1-1
SYMBOL TABLE

BELL  = 000007          CINIT = ****** G        DLOAD = ****** G        FIN2    000002R         PSEND = ****** G
BOOT    000034R         CR    = 000015          FIN     000000R         LF    = 000012          SBOOT = ****** G

. ABS.  000000      000
        000074      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  189 WORDS  ( 1 PAGES)
DYNAMIC MEMORY:  3162 WORDS  ( 12 PAGES)
ELAPSED TIME:  00:00:01
BOOT,BOOT=BOOT
```

APPENDIX D
BOOT23

0123456789012345678 ** RSX-11M V03 ** 20-APR-82 15:49:23 DB0:[212,10]BBBB123.LST;3 0123456789012345678
0123456789012345678 ** RSX-11M V03 ** 20-APR-82 15:49:23 DB0:[212,10]BBBB123.LST;3 0123456789012345678

D

```
BBBBBBBB      BBBBBB      BBBBBB     TTTTTTTTTT     222222      333333
BBBBBBBBB     BBBBBBB     BBBBBBB    TTTTTTTTTT     222222      333333
BB      BB    BB    BB    BB    BB       TT       22    22    33    33
BB      BB    BB    BB    BB    BB       TT       22    22    33    33
BB      BB    BB    BB    BB    BB       TT             22          33
BB      BB    BB    BB    BB    BB       TT             22          33
BBBBBBBBB     BB    BB    BB    BB       TT          22             33
BBBBBBBBB     BB    BB    BB    BB       TT          22             33
BB      BB    BB    BB    BB    BB       TT       22                33
BB      BB    BB    BB    BB    BB       TT       22                33
BB      BB    BB    BB    BB    BB       TT    22          33    33    ....
BB      BB    BB    BB    BB    BB       TT    22          33    33    ....
BBBBBBBBB     BBBBBB      BBBBBB         TT    2222222222     333333     ....
BBBBBBBBB     BBBBBB      BBBBBB         TT    2222222222     333333     ....
```

D-2

```
LL            SSSSSSSS   TTTTTTTTTT      ;;;;      333333
LL            SSSSSSSS   TTTTTTTTTT      ;;;;      333333
LL          SS             TT           ;;;;    33    33
LL          SS             TT           ;;;;    33    33
LL          SS             TT                   33
LL          SS             TT                   33
LL            SSSSSS       TT           ;;;;       33
LL            SSSSSS       TT           ;;;;       33
LL                  SS     TT           ;;;;          33
LL                  SS     TT           ;;;;          33
LL                  SS     TT             ;;    33    33
LL                  SS     TT             ;;    33    33
LLLLLLLLLL  SSSSSSSS       TT           ;;        333333
LLLLLLLLLL  SSSSSSSS       TT           ;;        333333
```

0123456789012345678 ** RSX-11M V03 ** 20-APR-82 15:49:23 DB0:[212,10]BBBB123.LST;3 0123456789012345678
0123456789012345678 ** RSX-11M V03 ** 20-APR-82 15:49:23 DB0:[212,10]BBBB123.LST;3 0123456789012345678
0123456789012345678 ** RSX-11M V03 ** 20-APR-82 15:49:23 DB0:[212,10]BBBB123.LST;3 0123456789012345678

```
                                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                        ;
                                        ;AUTHOR:        ROBERT B. NORMOYLE
                                        ;               QUEST RESEARCH CORP
                                        ;               MCLEAN, VA
                                        ;
                                        ;DATE:          20-NOV-81
                                        ;
                                        ;
                                        ;PURPOSE:       THIS PROGRAM IS DESIGNED TO RUN ON THE LSI 11/23 TO
                                        ;       DOWNLOAD PROGRAMS VIA THE PARALLEL PORT.  AT PRESENT THIS
                                        ;       PROGRAM IS DOWNLOADED TO THE 11/23 VIA THE SERIAL BOOT
                                        ;       ROUTINE SBOOT.
                                        ;       THE PROGRAM LOADS THE DATA TRANSMITTED OVER THE PARALLEL LINK
                                        ;       AT LOCATION 5000. THE FIRST WORD TRANSMITTED SHOULD
                                        ;       BE THE NUMBER OF WORDS THAT WILL FOLLOW. WHEN TRANSMISSION
                                        ;       IS COMPLETED THE ROUTINE JUMPS TO LOCATION 5002 AND STARTS
                                        ;       EXECUTING CODE AT THIS POINT.
                                        ;PURPOSE:       THIS PROGRAM IS DESIGNED TO RUN ON TO// LSI 11/23 TO
                                        ;       DOWNLOAD PROGRAMS VIA TO// PARALLEL PORT.  AT PRESENT THIS
                                        ;       PROGRAM IS DOWNLOADED TO//TO TO// 11/23 VIA TO// SERIAL BOOT
                                        ;       ROUTINE SBOOT.
                                        ;               TO// PROGRAM LOADS TO// DATA TRANSMITTED OVER
                                        ;       TO// PARALLEL PORT AT LOCATION 5000. WO//N IT IS FINISO//D LOADING
                                        ;       DATA IT TO//N JUMPS TO LOCATION 5002. DATA BLOCK SIZE TO BE
                                        ;       DOWNLOADED IS INDICATED BY TO// FIRST TRANSMITTED WORD TO TO//
                                        ;       11/23. THIS WORD IS PLACED AT LOCATION 5000.
                                        ;
                                        ;
                                        ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
        005000                          BEGLD=5000              ;BEGLD=11/23 MEMORY LOCATION TO LOAD PROGRAM FROM "A" PROC
        167770                          DRVCSR=167770           ;DRVCSR=PARALLEL PORT CONTROL STATUS REGISTER
        167774                          DRVIN=DRVCSR+4          ;DRVIN=PARALLEL PORT TO RECEIVE DATA FROM
        000002                          UNCSR1=2                ;UNCSR1=OPTION ON DRV11
                                                .GLOBL  SCOUNT
000000  000064                          SCOUNT: EXIT-SCOUNT             ;SCOUNT=LINK REGISTER IF THIS PROGRAM WERE
                                                                        ;TO BE USED WITH COMM2.
000002  012700  005000                  START:  MOV     #BEGLD,R0       ;R0=BEGINNING LOCATION OF MEMORY TO RECEIVE DATA
000006  005037  167770                          CLR     @#DRVCSR        ;RESET CONTROL STATUS REGISTERS OF DRV
                                        ;
000012  105737  167770                  WAITA:  TSTB    @#DRVCSR        ;WAIT UNTIL REQUEST "A" RESET ON DRV
000016  100375                                  BPL     WAITA
                                        ;
000020  013720  167774                          MOV     @#DRVIN,(R0)+   ;MOV INPUT INTO MEMORY
000024  052737  000002  167770                  BIS     #UNCSR1,@#DRVCSR        ;INFORM HOST NEW WORD WAS RECEIVED
000032  105737  167770                  WAITHA: TSTB    @#DRVCSR        ;WAIT UNTIL REG "A" SET
000036  100775                                  BMI     WAITHA
000040  042737  000002  167770                  BIC     #UNCSR1,@#DRVCSR        ;HOST ACKNOWLEDGES HANDSHAKE
000046  005337  005000                          DEC     @#BEGLD         ;DECREMENT NUMBER OF WORDS TO TRANSMIT
000052  001357                                  BNE     WAITA   ;IF # WORDS DOES NOT EQUAL ZERO THEN DO AGAIN
000054  005037  167770                          CLR     @#DRVCSR        ;ELSE, CLEAR DRV AND JUMP TO STARTING
                                                                        ;;LOCATION OF PROGRAM
000060  000137  005002                          JMP     @#BEGLD+2
000064  000000                          EXIT:   HALT
        000001                                  .END
```

```
.MAIN.  MACRO M11    20-APR-82 15:49  PAGE 1-1
SYMBOL TABLE

BEGLD = 005000          DRVIN = 167774          UNCSR1= 000002          START   000002R         WAITHA  000032R
DRVCSR= 167770          EXIT    000064R         SCOUNT  000000RG        WAITA   000012R

. ABS.  000000      000
        000066      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  89 WORDS  ( 1 PAGES)
DYNAMIC MEMORY:  3162 WORDS  ( 12 PAGES)
ELAPSED TIME:  00:00:01
BOOT23,BOOT23=BOOT23
```

APPENDIX E
COMM2

```
0123456789012345678?    ** RSX-11M V03 **     20-APR-82    15:52:03    DB0:[212,10]COMM2.LST;27    0123456789012345678?
0123456789012345678?    ** RSX-11M V03 **     20-APR-82    15:52:03    DB0:[212,10]COMM2.LST;27    0123456789012345678?
0123456789012345678?    ** RSX-11M V03 **     20-APR-82    15:52:03    DB0:[212,10]COMM2.LST;27    0123456789012345678?
```

```
  CCCCCCCC     UUUUUU     MM       MM   MM       MM     222222
  CCCCCCCC     UUUUUU     MM       MM   MM       MM     222222
CC           UU      UU   MMMM   MMMM   MMMM   MMMM   22      22
CC           UU      UU   MMMM   MMMM   MMMM   MMMM   22      22
CC           UU      UU   MM  MM  MM   MM  MM  MM            22
CC           UU      UU   MM  MM  MM   MM  MM  MM            22
CC           UU      UU   MM      MM   MM      MM          22
CC           UU      UU   MM      MM   MM      MM          22
CC           UU      UU   MM      MM   MM      MM        22
CC           UU      UU   MM      MM   MM      MM        22
CC           UU      UU   MM      MM   MM      MM      22              ....
CC           UU      UU   MM      MM   MM      MM      22              ....
  CCCCCCCC     UUUUUU     MM      MM   MM      MM   2222222222         ....
  CCCCCCCC     UUUUUU     MM      MM   MM      MM   2222222222         ....
```

E-2

```
LL            SSSSSSSS   TTTTTTTTTT      ;;;;      222222     77777777
LL            SSSSSSSS   TTTTTTTTTT      ;;;;      222222     77777777
LL          SS               TT         ;;;;    22      22           77
LL          SS               TT         ;;;;    22      22           77
LL          SS               TT                 22      22           77
LL          SS               TT                 22      22           77
LL            SSSSSS         TT         ;;;;         22          77
LL            SSSSSS         TT         ;;;;         22          77
LL                  SS       TT         ;;;;       22          77
LL                  SS       TT         ;;;;       22          77
LL                  SS       TT           ;;     22          77
LL                  SS       TT           ;;     22          77
LLLLLLLLLL    SSSSSSSS       TT         ;;     2222222222   77
LLLLLLLLLL    SSSSSSSS       TT         ;;     2222222222   77
```

```
0123456789012345678?    ** RSX-11M V03 **     20-APR-82    15:52:03    DB0:[212,10]COMM2.LST;27    0123456789012345678?
0123456789012345678?    ** RSX-11M V03 **     20-APR-82    15:52:03    DB0:[212,10]COMM2.LST;27    0123456789012345678?
0123456789012345678?    ** RSX-11M V03 **     20-APR-82    15:52:03    DB0:[212,10]COMM2.LST;27    0123456789012345678?
```

```
                                        .TITLE  COMM2
                                        .GLOBL  SCOUNT,SBOOT,PSEND,CIN11
000000  002334                  ICOUNT: EXIT-ICOUNT
                                ;
000002  000000                          HALT
                                ;MEMORY ADDRESS FOR 11/23 LINK PORT
                        175610  RCSRM=175610
                        175612  RBUFM=RCSRM+2
                        175614  XCSRM=RCSRM+4
                        175616  XBUFM=RCSRM+6
                                ;
000004  175610          RCSR:   .WORD   RCSRM           ;RECEIVER CONTROL STATUS REGISTER
000006  175612          RBUF:   .WORD   RCSRM+2         ;RECEIVER BUFFER
000010  175614          XCSR:   .WORD   RCSRM+4         ;TRANSMITTER CONTROL STATUS REGISTER
000012  175616          XBUF:   .WORD   RCSRM+6         ;TRANSMITTER BUFFER REGISTER
                                ;
                                ;FOR BOTH XCSR AND RCSR A HIGH IN BIT 7 INDICATES THAT THE
                                ;DEVICE IS READY TO PERFORM ITS SPECIFIED FUNCTION.
                                ;
                                ;
                                ;
                                ;MEMORY ADDRESSES FOR THE COMPUTERS CONSOLE PORT
                        177560  RCSRC=177560
                        177562  RBUFC=RCSRC+2
                        177564  XCSRC=RCSRC+4
                        177566  XBUFC=RCSRC+6
                                ;
                                ;
                        000015  CR=15                    ;OCTAL VALUE OF CARRIAGE RETURN
                        000012  LF=12                    ;OCTAL VALUE OF LINE FEED
                                ;
                        175610  TERMIN=175610
                        177560  CONSOL=177560
                                ;
                                ;
000014  000016'          BUF:    .WORD   BUF+2
                                .BLKW   200
000416  000420'          YES:    .WORD   YES+2
000420  114  123  111    YES2:   .ASCIZ  /LSI 11 23 ON-LINE/
        040  061  061
        040  062  063
        040  117  116
        055  114  111
        116  105  000
000442  177777                  YEND:   .WORD   177777
000444  000446'                 NO:     .WORD   NO+2
000446  114  123  111    NO2:    .ASCIZ  /LSI 11 23 OFF-LINE    RETRY??/<BELL>
        040  061  061
        040  062  063
        040  117  106
        106  055  114
        111  115  105
        040  040  040
        040  122  105
        124  122  131
        077  077  007
```

```
      000504     000
                                        .EVEN
      000506  177777                NEND:   .WORD   177777
              000007                BELL=07
                                    ;
```

```
                                          ;       THE FOLLOWING IS CODE FOR SERIAL COMMUNICATIONS BETWEEN
                                          ;THE "A" OR "B" PROCESSOR AND THE TERMINAL LINE HOOKED
                                          ;UP TO THE 11/23.
                                          ;
                                          ;
                                          ;
                                          ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                          ;
                                          ;       SBOOT:  PRIMARY PURPOSE IS TO AUTOMATICALLY BOOT THE LSI 11/23
                                          ;
                                          ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                          ;
           000000                         NULL=00
           000057                         SLASH=57
    000510    067     066     067         DRV11A: .ASCIZ  /767772/<SLASH>
    000513    067     067     062
    000516    057     000
                                                  .EVEN
    000520    061     067     067         DRV11D: .ASCIZ  /177777/
    000523    067     067     067
    000526    000
                                                  .EVEN
                                                  .EVEN
    000530    062     060     060         BEGMEM: .ASCIZ  /2000/<SLASH>           ;BEG MEMORY LOCATION IN LSI
    000533    060     057     000
    000536    062     060     060         GO:     .ASCIZ  /2002G/
    000541    062     107     000
                                                  .EVEN
    000544                                INBUFF: .BLKW   100                     ;INPUT BUFFER
                                          ;
                                          ;
    000744 004767  000740                 SBOOT:  CALL    READY           ;MAKE O.D.T. RUN IN LSI 11/23
    000750 004767  000506                         CALL    TINIT           ;SELECT TERMINAL SERIAL PORT
                                          ;       JMP     SB2
                                          ;
                                          ;                       TURN ON DRV LIGHTS
    000754 012704  000510'                        MOV     #DRV11A,R4
    000760 004767  000656                         CALL    PCHRSE          ;OUTPUT ADDRESS OF DRV11
    000764 103767                                 BCS     SBOOT           ;RESTART IF ECHO ERROR
    000766 004767  000336                         CALL    TREADS          ;WAIT UNTIL H23 READY
    000772 012704  000520'                        MOV     #DRV11D,R4              ;OUTPUT DATA TO H23
    000776 004767  000640                         CALL    PCHRSE          ;
    001002 103760                                 BCS     SBOOT           ;RESTART IF ECHOE ERROR
    001004 004767  000320                         CALL    TREADS          ;WAIT UNTIL H23 READY
    001010 004767  000120                         CALL    PCH
    001014 004767  000310                         CALL    TREADS          ;WAIT UNTIL H23 READY
                                          ;
                                          ;               OPEN STARTING LOCATION OF PROGRAM
    001020                                SB2:
    001020 012704  000530'                        MOV     #BEGMEM,R4      ;R4=FIRST MEM LOCATION IN LSI TO LOAD
    001024 004767  000612                         CALL    PCHRSE          ;PRINT STRING
    001030 103745                                 BCS     SBOOT           ;RESTART IF ECHO ERROR
                                          ;
                                          ;
    001032 016702  000000G                        MOV     SCOUNT,R2       ;R2=# OF BYTES TO TRANSFER TO LSI
    001036 006202                                 ASR     R2              ;R2=# OF WORDS TO TRANSFER TO LSI
    001040 012703  000000G                        MOV     #SCOUNT,R3      ;R3=FIRST WORD IN PDP 11 TO TRANSFER
```

```
                        ;
                                                        ;WAIT UNTIL LSI IS READY
001044  012704  000544'          MOV     #INBUFF,R4
001050  012701  000544'          MOV     #INBUFF,R1     ;READ LSI OUTPUT INTO INBUFF
001054  004767  000250    SBOOT1: CALL    TREADS         ;READ DATA UNTIL TIMED OUT ON A CHAR
001060  004767  001110          CALL    OCTASC         ;CONVERT OCT WORD TO ASCII STRING
001064  004767  000552          CALL    PCHRSE         ;PRINT CHAR STRING W/TIMED WAIT FOR ECHO
001070  103725                  BCS     SBOOT          ;RESTART BOOT IF ECHO ERROR
                        ;
001072  004767  000070          CALL    PLF            ;PRINT LINE FEED
                        ;
001076  005203                  INC     R3
001100  005203                  INC     R3             ;R3=NEXT WORD IN PDP MEM TO TRANSFER TO LSI
001102  077214                  SOB     R2,SBOOT1      ;IF R2=0 THEN DONT OUTPUT NEXT WORD
                        ;
                        ;               CLOSE MEMORY IN LSI
001104  004767  000024          CALL    PCR            ;PRINT CARRIAGE RETURN
                        ;
                        ;
                        ;               RUN PROGRAM
001110  004767  000214          CALL    TREADS         ;WAIT ON LSI
001114  012704  000536'          MOV     #GO,R4         ;PRINT 5002G
001120  004767  000516          CALL    PCHRSE
                        ;
                        ;               CLEAN UP
001124                          ENABLE  @#TERMIN
001132  000207                  RETURN
                        ;
                        ;
                        ;
                        ;
                        ;
001134                  PCR:            ; PRINT CARRIAGE RETURN
001134                          PUSH    R4
001136  012700  000015          MOV     #CR,R0
001142  004767  000250          CALL    PCHARE         ;PRINT CARRIAGE RETURN
001146  122700  000015          CMPB    #CR,R0         ;WAS CR ECHOED BACK
001152                          POP     R0             ;RESTORE R0
001154  001002                  BNE     PCR1           ;NO, SET CARRY FLAG
001156  000241                  CLC                    ;CLEAR CARRY FLAG
001160  000207                  RETURN                 ;RETURN
001162  000261          PCR1:   SEC                    ;SET CARRY FLAG
001164  000207                  RETURN                 ;RETURN
                        ;
                        ;
                        ;
                        ;               PRINT LINE FEED
001166                  PLF:    PUSH    R0             ;SAVE R0
001170  012700  000012          MOV     #LF,R0         ;R0=LINE FEED
001174  004767  000216          CALL    PCHARE         ;PRINT LINE FEED
001200  022700  000012          CMP     #LF,R0         ;WAS LF ECHOED BACK
001204                          POP     R0             ;RESTORE R0
001206  001002                  BNE     PLF1           ;NO, SET CARRY FLAG
001210  000241                  CLC                    ;YES,CLEAR CARRY FLAG
001212  000207                  RETURN                 ;RETURN
001214  000261          PLF1:   SEC                    ;SET CARRY FLAG
001216  000207                  RETURN                 ;RETURN
```

```
                                        ;
                                        ;
                                        ;       RCHAR WAITS FOR A CHARACTER FROM THE TERMINAL
                                        ;       LINE, AND PLACES CHARACTER INTO REGISTER R0
                                        ;
001220  105777  176560       RCHAR:  TSTB    @#RCSR          ;IS A NEW CHAR IN BUFFER
001224  100375                  BPL     RCHAR           ;NO, WAIT FOR NEW CHAR
001226  117700  176554          MOVB    @#RBUF,R0               ;YES, MOVE NEW CHAR INTO
                                                        ;R0
001232  000207                  RTS     PC              ;RETURN
                                        ;
                                        ;
                                        ;       RCHARE READS NEW CHAR IN @RBUF AND ECHOS
                                        ;       CHAR BACK TO ORIGINATING DEVICE
                                        ;
001234  004767  177760       RCHARE: JSR     PC,RCHAR        ;READ NEW CHAR INTO R0
001240  004767  000136          JSR     PC,PCHAR        ;PRINT R0 TO 11/23
001244  000207                  RTS     PC              ;RETURN
                                        ;
                                        ;
                                        ;       TRCHAR DOES THE SAME AS RCHAR BUT RETURNS TO
                                        ;       CALLING ROUTINE WITH CARRY FLAG IF NO NEW CHAR
                                        ;       AVAILABLE IN THE TIME TO CYCLE THROUGH WAIT
                                        ;       LOOP 65536 TIMES
                                        ;
001246                  TRCHAR:                         ;CLEAR CARRY FLAG
001246  010446              MOV     R4,-(SP)        ;PUSH R4
001250  005004              CLR     R4              ;CLEAR R4
001252  105777  176526  TRC1:   TSTB    @#RCSR          ;IS A NEW CHAR IN @RBUF
001256  100404              BMI     TRC2            ;YES, GO TO TRC2
001260  077404              SOB     R4,TRC1         ;NO, DEC R4 IF R4=0
                                                        ; THEN  CUNTINUE
                                                        ; ELSE GOTO TRC1
001262  000261              SEC                     ;TIMED OUT, SET CARRY FLAG
001264  012604              MOV     (SP)+,R4        ;POP R4
001266  000207              RTS     PC              ;RETURN
                                        ;
001270  117700  176512  TRC2:   MOVB    @#RBUF,R0               ;R0=CHAR
001274  042700  000200          BIC     #200,R0         ;CLEAR PARITY BIT
001300  120027  000032              CMPB    R0,#32                  ;WAS INPUT ^Z?
001304  001403              BEQ     TRC3            ;YES, ABORT PROGRAM
001306  000241              CLC                     ;CLEAR CARRY-ECHOE RECEIVED
001310  012604              MOV     (SP)+,R4        ;POP R4
001312  000207              RTS     PC              ;RETURN
001314                  TRC3:   ENABLE  @#TERMINS                       ;LET OPERATING SYSTEM CLEAN UP
001322                          EXITSS                  ;EXIT VIA EMULATOR TRAP
                                        ;
                                        ;
                                        ;
                                        ;;;;;;;;;
                                        ;TREADS         READ STRING UNTIL A CHAR READ TIMED OUT
                                        ;;;;;;;;;
                                        ;
                                        ;
                                        ;               R1=DESTINATION BUFFER LOCATION
```

```
                                        ;
                                        ;
001330                                  TREADS:
001330                                          PUSH    R0,R1,R2        ;SAVE R0,R1,R2
001336  012702  000400                  TRDS1:  MOV     #400,R2         ;PERMIT UPTO 256 CHAR TO BE READ
001342  004767  177700                          CALL    TRCHAR          ;READ CHAR
001346  103404                                  BCS     TRDS2           ;EXIT IF CARRY SET
001350  005302                                  DEC     R2              ;R2=NUMBER OF CHAR THAT MAY BE READ
001352  001402                                  BEQ     TRDS2           ;IF R2=0, THEN BUFFER FULL
001354  110021                                  MOVB    R0,(R1)+        ;SAVE CHAR BYTE
001356  000767                                  BR      TRDS1           ;READ CHAR UNTIL TIMED OUT
001360                                  TRDS2:  POP     R2,R1,R0        ;RESTORE REGISTERS
001366  000207                                  RETURN
                                        ;       TIMED READ CHAR WITH UNTIMED ECHOE
                                        ;
001370  004767  177652                  TRCHRE: JSR     PC,TRCHAR
001374  004767  000002                          JSR     PC,PCHAR
001400  000207                                  RTS     PC                      ;RETURN
```

```
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;       PCHAR PRINTS THE CONTENTS OF R0 WHEN
                                        ;       BIT 7 OF @XCSR IS HIGH
                                        ;       THIS BIT INFORMS US THAT THE XMITTER IS READY
                                        ;       FOR A NEW WORD
                                        ;
001402  105777  176402          PCHAR:  TSTB    @XCSR           ;IS XMITTER READY
001406  100375                          BPL     PCHAR           ;NO, WAIT
001410  010077  176376                  MOV     R0,@XBUF                ;YES, WRITE NEW WORD
001414  000207                          RTS     PC              ;RETURN
                                        ;
                                        ;
                                        ;       PCHAR PRINTS R0 AND WAITS FOR AN ECHO FROM
                                        ;       DEVICE. ECHOE IS PLACED IN R0
                                        ;
001416  004767  177760          PCHARE: JSR     PC,PCHAR        ;PRINT CHAR
001422  004767  177620                  JSR     PC,TRCHAR               ;WAIT FOR ECHOE
001426  000207                          RTS     PC                      ;RETURN
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;       THE NEXT SECTION IS SIMILIAR TO THE PREVIOUS
                                        ;       EXCEPT THAT THE CONTROLLED DEVICE IS THE CONSOLE RATHER
                                        ;       THAN THE 11/23
```

```
                                        ;
                                        ;
                                        ;
                                        ;       THE FOLLOWING IS CODE USED TO SET UP MEMORY
                                        ;       TO DRIVE SPECIFICIED DEVICE.
                                        ;
                                        ;       SET UP TO DRIVE CONSOLE
                                        ;
001430  012767  177560  176346  CINIT:  MOV     #RCSRC,RCSR
001436  012767  177562  176342          MOV     #RBUFC,RBUF
001444  012767  177564  176336          MOV     #XCSRC,XCSR
001452  012767  177566  176332          MOV     #XBUFC,XBUF
001460  000207                          RTS     PC                      ;RETURN
                                        ;
                                        ;
                                        ;
                                        ;       SET UP MEMORY TO DRIVE TERMINAL
                                        ;
001462  012767  175610  176314  TINIT:  MOV     #RCSRM,RCSR
001470  012767  175612  176310          MOV     #RBUFM,RBUF
001476  012767  175614  176304          MOV     #XCSRM,XCSR
001504  012767  175616  176300          MOV     #XBUFM,XBUF
001512  000207                          RTS     PC                      ;RETURN
                                        ;
                                        ;
                                        ;
```

```
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;       THE FOLLOWING PROGRAMS ARE USED TO PERFORM
                                ;       I/O ON STRINGS OF DATA
                                ;
                                ;
                                ;
                                ;       READSE  READS A STRING OF DATA AND
                                ;        ECHOS EACH CHARACTER AS IT IS READ
                                ;
                                ;
                                ;       ON INPUT:
                                ;               R1=DESTINATION BUFFER OF INPUT CHAR
                                ;               R2=# OF CHAR TO READ
                                ;
                                ;       ON OUTPUT:
                                ;               THE BUFFER THAT R1 POINTS TO IS FILLED WITH N=R2
                                ;               CHARACTERS.
                                ;
001514                          READSE:
001514                                  PUSH    R0,R1,R2        ;SAVE REGISTERS
001522  004767  177506          RSE1:   CALL    RCHARE          ;READ CHAR AND ECHO
001526  010021                          MOV     R0,(R1)+        ;SAVE CHAR
001530  077204                          SOB     R2,RSE1
                                        ;IF R2.NEQ.0 THEN READ NEXT CHAR
001532                                  POP     R2,R1,R0        ;RESTORE REGISTERS
001540  000207                          RETURN
                                ;
                                ;
                                ;

                                ;
                                ;       TREADS  READS A STRING OF DATA WITHOUT
                                ;       ECHO.
                                ;               EACH CHARACTER HAS A TIME OUT LIMIT
                                ;
                                ;       ON INPUT:
                                ;               R1=BUFFER BEGINNING LOC FOR INPUT CHAR
                                ;               R2=NUMBER OF CHAR TO ATTEMPT TO READ
                                ;
                                ;       ON OUTPUT:
                                ;               THE BUFFER THAT R1 POINTS TO IS FILLED WITH N=R2
                                ;               CHARACTERS.
                                ;
001542                          TREDDS:
001542                                  PUSH    R0,R1,R2        ;SAVE REGISTERS
001550  004767  177472          TRS2:   CALL    TRCHAR          ;READ CHAR
001554  010021                          MOV     R0,(R1)+        ;SAVE CHAR
001556  077204                          SOB     R2,TRS2         ;IF R2.NEQ.0 THEN READ NEXT CHAR
001560                                  POP     R2,R1,R0        ;RESTORE REGISTERS
001566  000207                          RETURN          ;RETURN
                                ;
                                ;
```

```
                                        ;
                                        ;
                                        ;
                                        ;       PS      PRINTS A STRING OF DATA TO THE OUTPUT DEVICE
                                        ;
                                        ;       ON INPUT:
                                        ;               R1=BEGINNING OF BUFFER TO BE PRINTED
                                        ;               R2=NUMBER OF CHAR IN BUFFER
                                        ;
001570                                  PS:
001570                                          PUSH    R0,R1,R2                ;SAVE REGISTERS
001576  012100                          PS2:    MOV     (R1)+,R0                ;R0=CHAR TO BE PRINTED
001600  004767  177576                          CALL    PCHAR
001604  077204                                  SUB     R2,PS2                  ;IF R2=0 THEN EXIT
001606                                          POP     R2,R1,R0                ;RESTORE REGISTERS
001614  000207                                  RETURN
                                        ;
                                        ;
                                        ;       PSEND   PRINTS A STRING OF DATA TO OUTPUT DEVICE
                                        ;               UNTIL IT FINDS A CHAR WITH BIT 7 HIGH
                                        ;
                                        ;       ON INPUT:
                                        ;               R1=BEG OF OUTPUT BUFFER
                                        ;
001616                                  PSEND:  PUSH    R0,R1
001622  112100                          PSEND1: MOVB    (R1)+,R0
001624  004767  177552                          CALL    PCHAR
001630  105700                                  TSTB    R0
001632  100373                                  BPL     PSEND1
001634                                          POP     R1,R0
001640  000207                                  RETURN
                                        ;
                                        ;
                                        ;;;;;;;;;
                                        ;PCHRSE                 PRINT STRING UNTIL NULL OR UNTIL ECHO ERROR
                                        ;;;;;;;;;
                                        ;
                                        ;
                                        ;               R4=BEGINNING OF STRING TO BE ECHOED
                                        ;
                                        ;
                                        ;
001642                                  PCHRSE:
001642                                          PUSH    R4              ;SAVE R4
001644  112400                          PSE1:   MOVB    (R4)+,R0        ;R0=ASCII BYTE TO TRANSMIT
001646  004767  177544                          CALL    PCHARE          ;CALL PRINT CHARACTER
001652  103413                                  BCS     PSE2            ;IF NO ECHO, THEN EXIT PROGRAM W/CARRY FLAG SET
                                        ;
001654  042700  177600                          BIC     #177600,R0
001660  126400  177777                          CMPB    -1(R4),R0       ;IF PRINT CHAR = ECHOE CHAR
001664  001006                                  BNE     PSE2            ;THEN CONTINUE
                                        ;
001666  122714  000000                          CMPB    #0,(R4)         ;IS NEXT CHAR END OF STRING
001672  001364                                  BNE     PSE1            ;NO, PRINT NEXT CHARACTER
001674  000241                                  CLC                     ;CLEAR CARRY FLAG
001676                                          POP     R4              ;RESTORE R4
001700  000207                                  RETURN
```

```
528 001702  000261                  PSE2:   SEC                     ;SET CARRY FLAG---ECHO ERROR
529 001704                                  POP     R4              ;RESTORE R4
530 001706  000207                          RETURN                  ;RETURN
```

```
                                ;       THE FOLLOWING PROGRAM CHECKS TO MAKE SURE
                                ;       THAT THE 11/23 IS PROPERLY HOOKED UP TO RECEIVE
                                ;       AND/OR TRANSMIT DAT TO THE 11/45 OR 11/55. WHILE
                                ;       OUT IS RUNNING ON THE 11/23, A 'CR' INPUT SHOULD CAUSE
                                ;       THE 11/23 TO ECHO BACK A CR,?,LF,@ STRING OF CHARACTERS.
                                ;       THIS PROGRAMS OUTPUTS THE CR AND WAITS FOR THE 5 CHAR
                                ;       RETURN FROM THE 11/23. IF THE INPUT IS DIFFERENT THEN
                                ;       THAT SPECIFIED ABOVE THEN THE ROUTINE LOOPS BACK TO THE
                                ;       BEGINNING AND PROCEEDS UNTIL 11/23 OUTPUTS CORRECT STRING
                                ;
                                ;
                                ;
001710                          READY:
001710                                  DISABLE @#TERMIN
001716  016701  176072                  MOV     BUF,R1          ;R1=BEGINNING OF BUFFER
001722  012700  000020                  MOV     #20,R0          ;CLEAR BUFFER
001726  005021                  READY2: CLR     (R1)+
001730  077002                          SOB     R0,READY2
                                ;
001732  016701  176056                  MOV     BUF,R1
001736  012700  000015                  MOV     #CR,R0                  ;R0=ASCII CARRIAAGE RETURN
001742  004767  177462                  CALL    CINIT           ;OUTPUT CR TO CONSOLE
001746  004767  177430                  CALL    PCHAR
001752  004767  177504                  CALL    TINIT           ;OUTPUT TO TERMINAL
001756  004767  000132                  CALL    BREAK           ;OUTPUT BREAK
001762  004767  177414                  CALL    PCHAR           ;OUTPUT CR
001766  012702  000004                  MOV     #4,R2           ;READ 4 CHAR FROM TERM PORT
001772  004767  177332                  CALL    TREADS
001776  004767  177426                  CALL    CINIT           ;PREPARE FOR CONSOLE COMMUNICATION
                                        ;
002002  016701  176006                  MOV     BUF,R1
002006  012702  000020                  MOV     #20,R2          ;R2=BUFFER SIZE
002012  142711  000200          READY3: BICB    #200,(R1)               ;MASK OUT PARITY BIT
002016  122721  000100                  CMPB    #'@,(R1)+               ;IS THERE AN @ IN INPUT
002022  001423                          BEQ     RDYYES          ;BRANCH IF "@" FOUND
002024  077206                          SOB     R2,READY3       ;CHECK 20 OCT CHAR
                                ;
002026  000730                          BR      READY           ;LOOP IN READY UNTIL 11 23 ON LINE
                                ;
                                ;
                                ;
                                ;
                                ;
                                ;       THIS SECTION OF CODE IS NOT USED WHEN THE PREVIOUS INSTRUC-
                                ;       TION IS          BR      READY.
                                ;
002030  016701  176410                  MOV     NO,R1           ;PRINT OFF-LINE MESSAGE
002034  004767  177556                  CALL    PSEND
002040  004767  177154                  CALL    RCHAR           ;WAIT FOR RESPONSE TO MESSAGE
002044  142700  000200                  BICB    #200,R0         ;CLEAR PARITY BIT
002050  122700  000131                  CMPB    #'1,R0          ;IS INPUT NEGATIVE
002054  001715                          BEQ     READY           ;YES, TRY "READY" AGAIN
                                ;               ENABLE TERMINAL AND EXIT
002056                                  ENABLE  @#TERMIN
002064                                  EXIT$S
                                ;
                                ;
```

```
                                        ;
                                        ;
002072  016701  176320          RDYYES: MOV     YES,R1          ;R1=BEGIN OF MESSAGE
002076  004767  177514                  CALL    PSEND
002102  004767  177060                  CALL    PLF
002106  004767  177022                  CALL    PCR             ;PRINT CR & LF
002112  000207                          RETURN                  ;RETURN
                                        ;
                                        ;
                                        ;
                                        ;
002114                          BREAK:
002114                                  PUSH    R0,R1
002120  005000                          CLR     R0
002122  012701  000010                  MOV     #10,R1
002126  052777  000001  175654          BIS     #1,@XCSR        ;SET BIT ZERO OF DEVICE
002134  077001                  BR2:    SOB     R0,BR2          ;WAIT
002136  077102                          SOB     R1,BR2
002140  042777  000001  175642          BIC     #1,@XCSR                ;RESET BIT ZERO
002146  000240                  BR3:    NOP
002150  000240                          NOP
002152  077003                          SOB     R0,BR3                  ;WAIT UNTIL TRANS LINE CLEAR
002154                                  POP     R1,R0
002160  000207                          RETURN
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;       CMPS    COMPARES STR POINTED TO BY R0
                                        ;       TO SEE IF IT IS EQUIVALENT TO STRING POINTED
                                        ;       TO BY R1. IF THEY ARE EQUIV, THEN RETURN WITH
                                        ;       EQUIV FLAG SET. IF NOT EQUIV THEN RETURN WITH
                                        ;       NEQUIV FLAG SET.  THE NUMBER OF WORDS TO TEST
                                        ;       IS IN R2
002162                          CMPS:
002162  022021                  CMP1:   CMP     (R0)+,(R1)+             ;IS (R0)=(R1)
002164  001401                          BEQ     CMP2                    ;YES, GOTO DEC COUNTER
002166  000207                          RETURN                          ;NO , RETURN
002170  077204                  CMP2:   SOB     R2,CMP1         ;DEC R2. IF R2.NE.0 THEN CMP1
002172  000207                          RETURN                  ;ELSE RETURN
                                        ;
                                        ;
                                        ;
                                        ;;;;;;;;
                                        ;OCTASC                 OCTAL TO ASCII CONVERSION
                                        ;;;;;;;;
                                        ;
                                        ;
                                        ;       R3=LOCATION FO WORD TO BE CONVERTED
                                        ;       R4=LOCATION TO STORE ASCII STRING
                                        ;
002174                          OCTASC: PUSH    R0,R1,R2,R3,R4          ;SAVE REGISTERS
002200  012702  000006                  MOV     #6,R2                   ;R2= NUMBER OF CHAR+1/WORD
002204  005001                          CLR     R1                      ;CLEAR WORKING REGISTER
002206  011300                          MOV     (R3),R0                 ;R0=WORD TO BE CONVERTED
002210  000412                          BR      OCTA1                   ;CONVERT R1 ORDER BIT
002212  062701  000060          OCTA2:  ADD     #60,R1                  ;CONVERT NIBBLE TO ASCII
```

```
000646 002224  110124                    MOVB    R1,(R4)+                ;SAVE ASCII WORD
000647 002226  005302                    DEC     R2                      ;DECREMENT ASCII CHAR COUNTER
000648 002230  001410                    BEQ     OCTA3                   ;EXIT IF 6 CHAR  CONVERTED
000649                                ;
000650 002232  005001                    CLR     R1                      ;R1=0
000651 002234  006100                    ROL     R0
000652 002236  006101                    ROL     R1
000653 002240  006100                    ROL     R0
000654 002242  006101                    ROL     R1
000655 002244                         OCTA1:
000656 002244  006100                    ROL     R0
000657 002246  006101                    ROL     R1                      ;R1= NEG OCTAL NIBBLE
000658 002250  000763                    BR      OCTA2
000659 002252                         OCTA3:
000660 002252  105014                    CLRB    (R4)            ;CLEAR BYTE TO INDICATE LAST CHAR
000661 002254                            POP     R4,R3,R2,R1,R0  ;RESTORE REGISTERS
000662 002266  000207                    RETURN                  ;RETURN
000663 002270                         TEST1:
000664 002270  012700  000060             MOV     #60,R0
000665 002274  004767  177162             CALL    TINIT
000666 002300  005001                     CLR     R1
000667 002302  004767  177074         TEST1A: CALL    PCHAR
000668 002306  077103                     SOB     R1,TEST1A
000669 002310  000240                     NOP
000670 002312  000207                     RTS     PC
000671                                ;
000672                                ;
000673                                ;
000674                                ;
000675                                ;
000676                                ;
000677                                ;
000678                                ;               TEST PROGRAM FOR PCHARE & PCHAR & TRCHAR
000679                                ;
000680 002314                         TPPT:
000681 002314  012704  000100             MOV     #100,R4
000682 002320  012700  000100             MOV     #'@,R0
000683 002324  004767  177066         TPPT1:  CALL    PCHARE
000684 002330  077403                     SOB     R4,TPPT1
000685 002332  000000                     HALT
000686                                ;
000687 002334  000240                 EXIT:   NOP
000688         000001                         .END
```

```
COMM2   MACRO M11    20-APR-82 15:51  PAGE 6-3
SYMBOL TABLE

BEGMEM  000530R      MPOP  = 000005      PS      001570R      READY2  001726R      TRC1    001252R
BELL  = 000007       MPUSH = 000000      PSEND   001616RG     READY3  002012R      TRC2    001270R
BREAK   002114R      MEND    000506R     PSEND1  001622R      RSE1    001522R      TRC3    001314R
BR2     002134R      NU      000444R     PSE1    001644R      SBOOT   000744RG     TRDS1   001336R
BR3     002146R      NU2     000446R     PSE2    001702R      SBOOT1  001054R      TRDS2   001360R
BUF     000014R      NULL  = 000000      PS2     001576R      SB2     001020R      TREADS  001330R
CINIT   001430RG     OCTASC  002174R     RMUF    000006R      SCOUNT= ****** G     TREDDS  001542R
CMPS    002162R      OCTA1   002244R     RBUFC = 177562       SLASH = 000057       TRS2    001550R
CMP1    002164R      OCTA2   002220R     RBUFM = 175612       TCOUNT  000000R      XBUF    000012R
CMP2    002170R      OCTA3   002252R     RCHAR   001220R      TERMIN= 175610       XBUFC = 177566
CONSOL= 177560       PCHAR   001402R     RCHARE  001234R      TEST1   002270R      XBUFM = 175616
CR    = 000015       PCHARE  001416R     RCSR    000004R      TEST1A  002302R      XCSR    000010R
DRV11A  000510R      PCHRSE  001642R     RCSRC = 177560       TINIT   001462R      XCSRC = 177564
DRV11D  000520R      PCR     001134R     RCSRM = 175610       TPPT    002314R      XCSRM = 175614
EXIT    002334R      PCR1    001162R     RDYYES  002072R      TPPT1   002326R      YEND    000442R
GO      000536R      PLF     001166R     READSE  001514R      TRCHAR  001246R      YES     000416R
INBUFF  000544R      PLF1    001214R     READY   001710R      TRCHRE  001370R      YES2    000420R
LF    = 000012

. ABS.  000000      000
        002336      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  1278 WORDS  ( 5 PAGES)
DYNAMIC MEMORY:  3162 WORDS  ( 12 PAGES)
ELAPSED TIME:  00:00:10
COMM2,COMM2=COMM2
```

APPENDIX F
TASM

```
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:54:09    DB0:[212,10]TASM.LST;2    01234567890123456789
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:54:09    DB0:[212,10]TASM.LST;2    01234567890123456789
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:54:09    DB0:[212,10]TASM.LST;2    01234567890123456789
```

```
TTTTTTTTTT     AAAAAA       SSSSSSSS   MM      MM
TTTTTTTTTT     AAAAAA       SSSSSSSS   MM      MM
    TT       AA      AA   SS           MMMM  MMMM
    TT       AA      AA   SS           MMMM  MMMM
    TT       AA      AA   SS           MM  MM  MM
    TT       AA      AA   SS           MM  MM  MM
    TT       AA      AA     SSSSSS     MM      MM
    TT       AA      AA     SSSSSS     MM      MM
    TT       AAAAAAAAAA           SS   MM      MM
    TT       AAAAAAAAAA           SS   MM      MM
    TT       AA      AA           SS   MM      MM        ....
    TT       AA      AA           SS   MM      MM        ....
    TT       AA      AA   SSSSSSSS     MM      MM        ....
    TT       AA      AA   SSSSSSSS     MM      MM        ....
```

```
LL            SSSSSSSS   TTTTTTTTTT       ;;;;       222222
LL            SSSSSSSS   TTTTTTTTTT       ;;;;       222222
LL          SS               TT           ;;;;     22      22
LL          SS               TT           ;;;;     22      22
LL          SS               TT                            22
LL          SS               TT                            22
LL            SSSSSS         TT           ;;;;           22
LL            SSSSSS         TT           ;;;;           22
LL                  SS       TT           ;;;;         22
LL                  SS       TT           ;;;;         22
LL                  SS       TT             ;;       22
LL                  SS       TT             ;;       22
LLLLLLLLLL   SSSSSSSS         TT           ;;       2222222222
LLLLLLLLLL   SSSSSSSS         TT           ;;       2222222222
```

```
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:54:09    DB0:[212,10]TASM.LST;2    01234567890123456789
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:54:09    DB0:[212,10]TASM.LST;2    01234567890123456789
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:54:09    DB0:[212,10]TASM.LST;2    01234567890123456789
```

```
C*****************************************************************************
C
C PROGRAM:      TASM.FTN
C
C AUTHOR:       ROBERT B. NORMOYLE
C               QUEST RESEARCH, INC
C               MCLEAN, VA 22101
C          PH:  821-3200
C
C PURPOSE:
C    SPECIFIC:  USED TO STRIP AWAY EXCESS  DATA IN A "TASK FILE"
C               WHICH IS GENERATED BY THE PDP 11 TASK BUILDER. THE
C               REMAINING DATA IS ASSEMBLY LANGUAGE CODE WHICH MAY
C               BE SAVED IN A FILE FOR FUTURE USE.THE MOTIVATION FOR
C               THIS PROGRAM IS TO BE ABLE TO CREATE PROGRAMS ON THE
C               PDP 11/70 AND SEND THE CODE TO A COMPUTER THAT DOES
C               NOT HAVE AN OPERATING SYSTEM TO ASSEMBLE,COMPILE
C               OR EDIT PROGRAMS.
C               THE DATA CREATED BY THE TASK BUILDER INCLUDES THE
C               OBJECT FILES WHICH MAY  RUN ON ANY PDP 11,WITH
C               OR WITHOUT AN OPERATING SYSTEM. IT ALSO INCLUDES
C               DATA WHICH IS ONLY NECESSARY FOR COMPUTERS THAT
C               HAVE AN OPERATING SYSTEM. THE FOLLOWING PROGRAM
C               IS USED TO CREATE A NEW FILE THAT CONTAINS ONLY
C               THE DATA REQUIRED TO RUN A PROGRAM ON A PDP 11
C               COMPUTER THAT DOES NOT HAVE AN OPERATING SYSTEM.
C               EXAMPLE: THE CTS SEEKER COMPUTERS PDP 11/23 AND
C                        PDP 11/34
C               THE CODE OF AN ASSEMBLY LANGUAGE FILE STARTS IN
C               RECORD FOUR (4) OF THE CORRESPONDING TASK FILE.
C
C    GENERAL:   USED TO EXAMINE ANY RECORD WHOSE BLOCK SIZE IS 512
C               BYTES. THE PROGRAM IS ALSO CAPABLE OF STORING THE
C               ENTIRE INPUT FILE IN ANOTHER FILE, OR  ANY PART
C               (OR PARTS) OF THE INPUT FILE IN ANOTHER FILE.
C
C       NOTE:   THE INPUT AND OUTPUT FILE RECORD SIZE MAY BE
C               CHANGED TO SUIT USER FILE BLOCK SIZE. IN THE
C               OPEN STATEMENTS THE SPECIFIED SIZE MUST BE
C               4 TIMES SMALLER THEN  THE BYTE SIZE OF EACH FILE
C               IN YOUR RECORD. WHEN CREATING A NEW TASK
C               FILE FOR THE ALTERED TASM.FTN PROGRAM, BE SURE
C                 THE OPTION:
C                      MAXBUFF = NBYTES
C               WHERE NBYTES IS THE NUMBER OF BYTES IN A RECORD,
C               OR 4 TIMES THE RECORDSIZE STATED IN THE FORTRAN
C               "OPEN" STATEMENTS.
C
C*****************************************************************************
0001            DIMENSION IB(9000),IOB(3000)
0002            DIMENSION INAME(15),LNAME(15)
C
C               DETERMINE WHICH FILE TO READ INTO INPUT BUFFER (IB)
C               AND OPEN IT.
C
C
```

```
0003     110     IRANGE=0
0004             WRITE(5,100)
0005     100     FORMAT('  ENTER FILE NAME.')
         C
0006             READ(5,120)(LNAME(J),J=1,15)
0007     120     FORMAT(15A2)
0008             LNAME(15)=0
         C
0009     5000    OPEN(UNIT=4,NAME=LNAME,FORM='UNFORMATTED',TYPE='OLD'
                 1       ,ASSOCIATEVARIABLE=INEXT,ACCESS='DIRECT'
                 2       ,RECORDSIZE=128,ERR=110)!       (128 DOUBLE WORDS/RECORD)
0010             TYPE 400
0011     400     FORMAT(' READ FROM RECORD (1-98)?',$)
0012             ACCEPT *,INEXT
0013             IF(INEXT.GE.99) GO TO 10000
0014             READ(4'INEXT,ERR=7001)(IB(N),N=1,256)!  READ ENTIRE RECORD (256 WORDS)
0015             GO TO 7000
         C
         C               IF ERROR IN READ THEN PRINT RECORD NUMBER
         C               AND ELEMENT NUMBER
         C
         C
0016     7001    WRITE(5,7100)N,INEXT
0017     7100    FORMAT(2X,' ERROR IN READ',5X,'LOCATION ',I3
                 1       ,'    RECORD ',I3)
         C
         C               ENDIF
         C
0018     7000    CLOSE(UNIT=4)
         C
         C               SPECIFY RECORD OF FILE TO EXAMINE
         C               SAVE ANY DATA IN FILE?
         C               INPUT LOCATION OF FIRST AND LAST WORD TO SAVE
         C
         C
0019             WRITE(5,2210)(IB(N),N=1,256)
0020     2210    FORMAT(8O8)
0021             TYPE 4100
0022     4100    FORMAT(' SAVE CODE (Y,N)?',$)
0023             ACCEPT 4550,IQUEST
0024             IF(IQUEST.EQ.'N')GO TO 5000
0025     4250    TYPE 4300
0026     4300    FORMAT(' STARTING LOCATION (1-256)=',$)
0027             ACCEPT *,IBEG
0028             TYPE 4400
0029     4400    FORMAT(' END LOCATION (1-256)=',$)
0030             ACCEPT *,IEND
         C
         C
         C
         C               WRITE SPECIFIED DATA TO OUTPUT FILE
         C
         C               READ ANOTHER RECORD?
         C               TRANSFER SAVED DATA TO OUTPUT BUFFER?
         C               OPEN ANOTHER INPUT FILE?
```

```
        C
        C
0031            DO 4450 N=IBEG,IEND
0032    4450    IOB(IRANGE+N-IBEG+1)=IB(N)
0033            IRANGE=IRANGE+(IEND-IBEG)+1!OUTPUT FILE BEG LOC=NEXT  WORD
0034            WRITE(5,2210)(IOB(N),N=1,IRANGE)
0035            TYPE 4500
0036    4500    FORMAT(' READ NEXT RECORD?',$)
0037            ACCEPT 4550,IQUEST
0038    4550    FORMAT(A2)
0039            IF(IQUEST.EQ.'Y') GO TO 5000
0040            TYPE 4600
0041    4600    FORMAT(' OUTPUT FILE TO DISK?',$)
0042            ACCEPT 4550,IQUEST
0043            IF(IQUEST.EQ.'Y')GO TO 9000
0044            TYPE 4700
0045    4700    FORMAT(' READ ANOTHER FILE?',$)
0046            ACCEPT 4550,IQUEST
0047            IF(IQUEST.EQ.'N')STOP
0048            GO TO 110!      OPEN ANOTHER FILE
        C
        C
        C
        C
        C               OUTPUT FILE TO DISK
0049    9000    TYPE 9100
0050    9100    FORMAT(' OUTPUT FILE NAME=',$)
0051            READ(5,120)(INAME(J),J=1,15)
0052            INAME(15)=0
0053            OPEN(UNIT=4,NAME=INAME,FORM='UNFORMATTED'
                1       ,RECORDSIZE=128)
0054            WRITE(4)(IOB(N),N=1,IRANGE)
0055            CLOSE(UNIT=4,ERR=9900)
0056            TYPE 9200
0057    9200    FORMAT(' READ ANOTHER FILE?',$)
0058            ACCEPT 4550,IQUEST
0059            IF(IQUEST.EQ.'Y')GO TO 110
0060    10000   STOP
0061    9900    TYPE *,' ERROR ON CLOSE',N
0062            STOP
0063            END
```

PROGRAM SECTIONS

NUMBER	NAME	SIZE		ATTRIBUTES
1	$CODE1	001736	495	RW,I,CON,LCL
2	$PDATA	000020	8	RW,D,CON,LCL
3	$IDATA	000574	190	RW,D,CON,LCL
4	$VARS	057012	12037	RW,D,CON,LCL

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
IBEG	I*2	4-057006	IEND	I*2	4-057010	INEXT	I*2	4-057000	IQUEST	I*2	4-057004	IRANGE	I*2	4-056774
J	I*2	4-056776	N	I*2	4-057002									

ARRAYS

NAME	TYPE	ADDRESS	SIZE		DIMENSIONS
IB	I*2	4-000000	043120	9000	(9000)
INAME	I*2	4-056700	000036	15	(15)
IUB	I*2	4-043120	013560	3000	(3000)
LNAME	I*2	4-056736	000036	15	(15)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
100'	3-000000	110	1-000020	120'	3-000026	400'	3-000032	2210'	3-000152
4100'	3-000156	4250	**	4300'	3-000204	4400'	3-000244	4450	**
4500'	3-000276	4550'	3-000324	4600'	3-000330	4700'	3-000362	5000	1-000142
7000	1-000414	7001	1-000346	7100'	3-000070	9000	1-001350	9100'	3-000412
9200'	3-000440	9900	1-001666	10000	1-001652				

FUNCTIONS AND SUBROUTINES REFERENCED

CLOSS OPENS

TOTAL SPACE ALLOCATED = 061564 12730

NO FPP INSTRUCTIONS GENERATED

TASM,TASM=TASM

APPENDIX G
TUDUM2.MAC

```
0123456789012345678    ** RSX-11M V03 **    20-APR-82    15:54:54    DB0:[212,10]TUDUM2.MAC;1    0123456789012345678
0123456789012345678    ** RSX-11M V03 **    20-APR-82    15:54:54    DB0:[212,10]TUDUM2.MAC;1    0123456789012345678
0123456789012345678    ** RSX-11M V03 **    20-APR-82    15:54:54    DB0:[212,10]TUDUM2.MAC;1    0123456789012345678
```

```
TTTTTTTTTT   UU      UU   DDDDDDDD     UU      UU   MM      MM     222222
TTTTTTTTTT   UU      UU   DDDDDDDDD    UU      UU   MM      MM     222222
    TT       UU      UU   DD      DD   UU      UU   MMMM  MMMM   22      22
    TT       UU      UU   DD      DD   UU      UU   MMMMM MMMMM  22      22
    TT       UU      UU   DD      DD   UU      UU   MM MMM MM          22
    TT       UU      UU   DD      DD   UU      UU   MM  M  MM          22
    TT       UU      UU   DD      DD   UU      UU   MM     MM        22
    TT       UU      UU   DD      DD   UU      UU   MM     MM        22
    TT       UU      UU   DD      DD   UU      UU   MM     MM      22
    TT       UU      UU   DD      DD   UU      UU   MM     MM      22
    TT       UU      UU   DD      DD   UU      UU   MM     MM    22          ....
    TT       UU      UU   DD      DD   UU      UU   MM     MM    22          ....
    TT       UUUUUUUUUU   DDDDDDDD     UUUUUUUUUU   MM     MM    2222222222  ....
    TT       UUUUUUUUUU   DDDDDDDD     UUUUUUUUUU   MM     MM    2222222222  ....
```

G-2

```
MM      MM     AAAAAA       CCCCCCCC       ;;;;         11
MM      MM     AAAAAA       CCCCCCCC       ;;;;         11
MMMM  MMMM    AA    AA     CC             ;;;;       1111
MMMM  MMMM    AA    AA     CC             ;;;;       1111
MM  MM  MM    AA    AA     CC                          11
MM  MM  MM    AA    AA     CC                          11
MM      MM    AA    AA     CC             ;;;;         11
MM      MM    AA    AA     CC             ;;;;         11
MM      MM    AAAAAAAAAA   CC             ;;;;         11
MM      MM    AAAAAAAAAA   CC             ;;;;         11
MM      MM    AA    AA     CC               ;;         11
MM      MM    AA    AA     CC               ;;         11
MM      MM    AA    AA       CCCCCCCC       ;;       111111
MM      MM    AA    AA       CCCCCCCC       ;;       111111
```

```
0123456789012345678    ** RSX-11M V03 **    20-APR-82    15:54:54    DB0:[212,10]TUDUM2.MAC;1    0123456789012345678
0123456789012345678    ** RSX-11M V03 **    20-APR-82    15:54:54    DB0:[212,10]TUDUM2.MAC;1    0123456789012345678
0123456789012345678    ** RSX-11M V03 **    20-APR-82    15:54:54    DB0:[212,10]TUDUM2.MAC;1    0123456789012345678
```

```
                00020
000E            00090 BH14      EQU     0EH
0033            00100 DSP       EQU     033H            ;DISP CHARC.

002B            00110 KBD       EQU     02BH            ;SCAN KEYBOA
RD
0046            00120 CIG       EQU     046H            ;MSG OUT
                00130 ;
                00140 ;
C000            00150           ORG     0C000H
C000 21FFCB     00152 START1    LD      HL,0CBFFH       ;STACK AT TO
P OF LRAM
C003 F9         00154           LD      SP,HL
C004 CD64C5     00160           CALL    INIT
C007 CD93C5     00170           CALL    INIT2
C00A CDAAC4     00190           CALL    OTDAT
C00D CD8FC4     00200           CALL    IOART
C010 CD8FC4     00210           CALL    IOART
C013 CDAAC4     00215           CALL    OTDAT
C016 CD3FC7     00220           CALL    BOOT
C019 CDCCC6     00300 START     CALL    SCKEX
C01C CDD5C6     00305           CALL    SCAN
C01F CDADC1     00310           CALL    SIGNON
                01000 ;         MAIN
C022 00         01010 MAIN      NOP
C023 CDA2C0     01020           CALL    ALLOC
C026 213AC3     01030           LD      HL,INPL
C029 CD1EC4     01040           CALL    OUTP
C02C CD37C4     01050           CALL    INPC
C02F CD2BC4     01060           CALL    OUTDRV
C032 CD54C4     01065           CALL    CR
C035 0630       01070           LD      B,48
C037 98         01080           SBC     A,B
C038 3202CB     01090           LD      (TSEL),A
C03B FE0A       01100           CP      10
C03D 2829       01105           JR      Z,SELA14
C03F FE01       01106           CP      1
C041 CC8BC0     01107           CALL    Z,DSEL14
C044 FE06       01108           CP      6
C046 CC8BC0     01109           CALL    Z,DSEL14
C049 FE15       01110           CP      21              ; (E)
C04B CAF9C0     01120           JP      Z,SOFF
C04E FE0C       01122           CP      12
C050 F222C0     01124           JP      P,MAIN
C053 FE00       01126           CP      0
C055 FA22C0     01128           JP      M,MAIN
C058 3A4FCB     01130           LD      A,(FH23)
C05B B7         01135           OR      A
C05C 3E01       01140           LD      A,1
C05E F262C0     01145           JP      P,MAIN1
C061 AF         01150           XOR     A
C062 324FCB     01155 MAIN1     LD      (FH23),A
C065 C322C0     01900           JP      MAIN
C068 3E0E       01912 SELA14    LD      A,0H14
C06A 3200CB     01915           LD      (CNTP),A
C06D CDACC6     01920           CALL    DSPLA
C070 3E01       01930           LD      A,1
C072 3202CB     01940           LD      (TSEL),A
C075 CD32C1     01960           CALL    SETL
C078 3E81       01962           LD      A,81H
C07A 324FCB     01964           LD      (FH23),A
C07D 2130CA     01965           LD      HL,MIARY
```

```
C083 012000      01967          LD      BC,20H
C086 EDB0        01968          LDIR
C088 C322C0      01970          JP      MAIN
C08B F5          01971 DSEL14   PUSH    AF
C08C AF          01972          XOR     A
C08D 3286CB      01973          LD      (FH23),A
C090 3A00CB      01974          LD      A,(CNTR)
C093 F5          01976          PUSH    AF
C094 3E0E        01979          LD      A,0H14
C096 3200CB      01982          LD      (CNTR),A
C099 CDA8C6      01985          CALL    DSEL
C09C F1          01988          POP     AF
C09D 3200CB      01991          LD      (CNTR),A
C0A0 F1          01994          POP     AF
C0A1 C9          01995          RET
C0A2 00          02000 ALLOC    NOP
C0A3 3E2F        02002          LD      A,47
C0A5 3204CB      02004          LD      (DUM),A
C0A8 21DDC5      02006          LD      HL,C0-10H
C0AB 2286CB      02008          LD      (DUM2),HL
C0AE 21B0C2      02020          LD      HL,MSG1
C0B1 CD1EC4      02025          CALL    OUTP
C0B4 CD54C4      02030          CALL    CR
C0B7 215BC2      02040          LD      HL,MSG01
C0BA CD1EC4      02050          CALL    OUTP
C0BD 21FEC5      02060          LD      HL,C1
C0C0 DD21FFC9    02065          LD      IX,CUMSEL-1
C0C4 AF          02070          XOR     A
C0C5 3207CB      02080          LD      (DUM3),A
C0C8 CDDEC0      02090 ALLOC1   CALL    ALLO1
C0CB CDDFC4      02100          CALL    ALLO
C0CE 3A07CB      02110          LD      A,(DUM3)
C0D1 3C          02120          INC     A
C0D2 3207CB      02130          LD      (DUM3),A
C0D5 060B        02140          LD      B,11
C0D7 B8          02150          CP      B
C0D8 20EE        02160          JR      NZ,ALLOC1
C0DA CD54C4      02280          CALL    CR
C0DD C9          02290          RET
C0DE 3A04CB      02400 ALLO1    LD      A,(DUM)
C0E1 3C          02410          INC     A
C0E2 3204CB      02420          LD      (DUM),A
C0E5 CD2BC4      02430          CALL    OUTDRV
C0E8 CDE6C5      02440          CALL    SPACE
C0EB 011100      02450          LD      BC,11H
C0EE 2A86CB      02460          LD      HL,(DUM2)
C0F1 09          02470          ADD     HL,BC
C0F2 2286CB      02480          LD      (DUM2),HL
C0F5 CD1EC4      02490          CALL    OUTP
C0F8 C9          02500          RET
C0F9 CD54C4      03000 SOFF     CALL    CR              ;SIGN OFF RO
UTINE
C0FC 215DC3      03010          LD      HL,MSG3
C0FF CD1EC4      03020          CALL    OUTP
C102 2A80CB      03030          LD      HL,(WTEMP)
C105 CD1EC4      03040          CALL    OUTP
C108 CD54C4      03050          CALL    CR
C10B 217BC3      03060          LD      HL,MSG4
C10E CD1EC4      03070          CALL    OUTP
C111 CDAAC4      03080          CALL    GIDAT
C114 3E00        03082          LD      A,0
C116 3205CB      03084          LD      (FLAG2),A
C119 C319C0      03090          JP      START
C11C CD2BC4      03210          CALL    OUTDRV
C11F E1          03300 NOAT     POP     HL
C120 21BAC3      03302          LD      HL,MSG5
```

```
C125 CD1EC4   03310          CALL    OUT
C126 217BC3   03320          LD      HL,MSG4
C129 CD1EC4   03330          CALL    OUTP
C12C CDAAC4   03332          CALL    OTDAT
C12F C319C0   03340          JP      START
C132 00       04000 SETL     NOP              ;ALLOCATED LINES
C133 C389C1   04005          JP      CKFL
C136 00       04007 SETL1    NOP
              04020 ;
              04030 ;SELECTION IN TSEL
C137 CDA8C6   04040          CALL    DSEL
C13A 2100CA   04050          LD      HL,COMSEL
C13D 3A02CB   04060          LD      A,(ISEL)
C140 4F       04063          LD      C,A
C141 0600     04065          LD      B,0
C143 09       04070          ADD     HL,BC
C144 7E       04072          LD      A,(HL)
C145 E6F0     04074          AND     0F0H
C147 47       04076          LD      B,A
C148 3A00CB   04080          LD      A,(CNTR)
C14B 80       04082          ADD     A,B
C14C 77       04090          LD      (HL),A
C14D CD54C4   04100          CALL    CR
C150 C9       04990          RET
C151 3E37     05000 URST     LD      A,37H
C153 D309     05010          OUT     (CTRL),A
C155 C9       05020          RET
C156 D9       05100 DELAY1   EXX
C157 110000   05105          LD      DE,0H
C15A 21F000   05110 D2       LD      HL,0F0H
C15D 13       05120          INC     DE
C15E ED52     05130          SBC     HL,DE
C160 20F8     05140          JR      NZ,D2
C162 D9       05150          EXX
C163 C9       05160          RET
C164 FE7F     05200 DELETE   CP      07FH
C166 2020     05210          JR      NZ,RET4
C168 2B       05220          DEC     HL
C169 3204CB   05222          LD      (DUM),A
C16C ED4384CB 05223          LD      (DUM1),BC
C170 2206CB   05224          LD      (DUM2),HL
C173 3E2F     05240          LD      A,47             ;/ CHAR.
C175 CD2BC4   05244          CALL    OUTDRV
C178 7E       05246          LD      A,(HL)
C179 CD2BC4   05250          CALL    OUTDRV
C17C 3A04CB   05262          LD      A,(DUM)
C17F ED4B84CB 05264          LD      BC,(DUM1)
C183 2A86CB   05266          LD      HL,(DUM2)
C186 3600     05267          LD      (HL),00H
C188 C9       05270 RET4     RET
C189 3A05CB   05300 CKFL     LD      A,(FLAG2)
C18C 0600     05310          LD      B,0
C18E B8       05320          CP      B
C18F C236C1   05330          JP      NZ,SETL1
C192 CD9BC1   05340          CALL    SWITCH
C195 C336C1   05360          JP      SETL1
C198 2130CA   06000 SWITCH   LD      HL,MAIN
C19B ED5B80CB 06010          LD      DE,(MTEMP)
C19F 012000   06020          LD      BC,20H
C1A2 EDB0     06040          LDIR
C1A4 CD93C5   06045          CALL    INIT2
C1A7 3E01     06050          LD      A,1
C1A9 3205CB   06060          LD      (FLAG2),A
C1AC C9       06100          RET
              07000 ;        SIGN ON ROUTINE
```

```
C1B0 CD93C4    07012          CALL    INIT2
C1B3 AF        07014          XOR     A
C1B4 3205CB    07016          LD      (FLAG2),A
C1B7 CD54C4    07020          CALL    CR
C1BA 2129C2    07030          LD      HL,MSG0
C1BD CD1EC4    07040          CALL    OUTP
C1C0 2103C3    07050          LD      HL,SGNON
C1C3 CD1EC4    07055          CALL    OUTP
C1C6 3A00CB    07060          LD      A,(CN1R)
C1C9 C630      07062          ADD     A,48
C1CB CD2BC4    07065          CALL    OUTDRV
C1CE CD54C4    07070          CALL    CR
C1D1 21F7C3    07075          LD      HL,SGNON1
C1D4 CD1EC4    07080          CALL    OUTP
C1D7 2130CA    07118          LD      HL,NTARY
C1DA CD11C4    07120 NALP     CALL    INPC1
C1DD CD2BC4    07123          CALL    OUTDRV          ;ECHO CHARC.

C1E0 CD64C1    07124          CALL    DELETE
C1E3 FE7F      07125          CP      07FH            ;CHECK FOR B
ACKSPACE
C1E5 28F3      07126          JR      Z,NALP
C1E7 FE0D      07129          CP      0DH             ;CHECK FOR C
.R.
C1E9 280E      07130          JR      Z,HEL
C1EB FE12      07132          CP      012H
C1ED CA0FC2    07134          JP      Z,REPTN
C1F0 FE15      07135          CP      015H
C1F2 CAADC1    07137          JP      Z,SIGNON
C1F5 77        07139          LD      (HL),A
C1F6 23        07140          INC     HL
C1F7 18E1      07145          JR      NALP
C1F9 00        07150 HEL      NOP
C1FA 3600      07152          LD      (HL),00H
C1FC CD54C4    07153          CALL    CR
C1FF 210AC4    07155          LD      HL,SGNON2
C202 CD1EC4    07158          CALL    OUTP
C205 2130CA    07160          LD      HL,NTARY
C208 CD1EC4    07165          CALL    OUTP
C20B CD54C4    07170          CALL    CR
C20E C9        07H00          RET
               08000 ;        MESSAGE AREA
C20F 22H6CB    08002 REPTN    LD      (DUM2),HL
C212 ED43H4CB  08003          LD      (DUM1),BC
C216 CD54C4    08004          CALL    CR
C219 2130CA    08006          LD      HL,NTARY
C21C CD1EC4    08008          CALL    OUTP
C21F ED4BB4CB  08010          LD      BC,(DUM1)
C223 2AH6CB    08011          LD      HL,(DUM2)
C226 C3DAC1    08015          JP      NALP
C229 2A        08020 MSG0:    DEFM    '***  BLACK HOLE TERMINAL CO
MMOD SYSTEM   ***'
C256 0D        08030          DEFB    0DH
C257 0A        08033          DEFB    0AH
C258 0D        08035          DEFB    0DH
C259 0A        08038          DEFB    0AH
C25A 00        08039          DEFB    00H
C25B 45        08040 MSG01    DEFM    'ENTER  COMP. LINE          N
H#       USER'
C283 0D        08050          DEFB    0DH
C284 0A        08055          DEFB    0AH
C285 2D        08060          DEFM    '-----  ----------          -
--       ----'
C2AD 0D        08070          DEFB    0DH
C2AE 0A        08075          DEFB    0AH
```

```
C2BD 2A        08200 MSG1:   DEFM    '**  TERMINAL/COMPUTER PREP
ENT ALLOCATION  ***'
C2DB 0D        08202         DEFB    0DH
C2DF 0A        08203         DEFB    0AH
C2E0 00        08205         DEFB    00H
C2E1 42        08210 BH      DEFM    'BH'
C2E3 00        08215         DEFB    00H
C2E4 20        08220 SP1     DEFM    '    '
C2E9 00        08230         DEFB    00H
C2EA 20        08240 SP2     DEFM    ' *** '
C2EF 00        08250         DEFB    00H
C2F0 20        08300 SETLM   DEFM    ' '
C2F1 4E        08310 BHOP    DEFM    'NOT ALLOCATED'
C2FE 00        08320         DEFB    00H
C2FF 20        08330 MSG2    DEFM    '   ENTER      COMPUTER LINE
'
C31A 0D        08340         DEFB    0DH
C31B 0A        08350         DEFB    0AH
C31C 20        08360         DEFM    '   -----      -------------
'
C337 0D        08370         DEFB    0DH
C338 0A        08380         DEFB    0AH
C339 00        08400         DEFB    00H
C33A 20        08410 INPL    DEFM    '   SELECTION NUMBER  <E> TO
 EXIT ?''
C35C 00        08420         DEFB    00H
C35D 43        08500 MSG3    DEFM    'CONNECTION MADE.  GOODBYE '

C377 00        08510         DEFB    0H
C378 48        08520 MSG4    DEFM    'HAVE A NICE DAY'
C387 0D        08530         DEFB    0DH
C388 0A        08540         DEFB    0AH
C389 00        08550         DEFB    0H
C38A 0D        08600 MSG5    DEFB    0DH
C38B 0A        08602         DEFB    0AH
C38C 4E        08604         DEFM    'NO ACTIVITY, SIGNING OFF'
C3A4 0D        08610         DEFB    0DH
C3A5 0A        08620         DEFB    0AH
C3A6 00        08630         DEFB    0H
               08800 ;;;;;;;;;; BOOT MSG.
C3A7 3F        08805 BOOT1   DEFM    '?????????? SYSTEM HAS REBOOT
ED. ??????????'
C3D0 0D        08810         DEFB    0DH
C3D1 0A        08815         DEFB    0AH
C3D2 00        08825         DEFB    00H
               08900 ;       SIGN ON
C3D3 48        08905 SGNON   DEFM    'HELLO.  YOU ARE ON TERMINAL
 LINE BH'
C3F6 00        08910         DEFB    00H
C3F7 59        08915 SGNUN1  DEFM    'YOUR NAME PLEASE'
C407 0D        08920         DEFB    0DH
C408 0A        08925         DEFB    0AH
C409 00        08930         DEFB    00H
C40A 48        08935 SGNUN2  DEFM    'HELLO '
C410 00        08940         DEFB    00H
C411 CD0AC4    09050 INPC1   CALL    INSER           ;GET SERIAL
C414 B7        09055         OR      A
C415 28FA      09060         JR      Z,INPC1
C417 0623      09065         LD      B,35            ;ASCII FOR #

C419 B8        09070         CP      B               ;CHECK FOR #

C41A 28F5      09075         JR      Z,INPC1         ;GET NEXT IN
PUT CHAR IF #
C41C 00        09077         DEFB    00H
```

```
[illegible] # ENTRY
C41E 7E        09100 OUTP    LD      A,(HL)
C41F FE00      09110         CP      00H
C421 2807      09120         JR      Z,RET
C423 CD2BC4    09130         CALL    OUTDRV          ;DISP CHARC.

C426 23        09140         INC     HL
C427 C31EC4    09150         JP      OUTP
C42A C9        09200 RET     RET                     ;RETURN
C42B 00        09300 OUTDRV  NOP     ;CALL   033H            ;ROM
 DISP CHARC.
C42C CD56C1    09305         CALL    DELAY1
C42F 4F        09310         LD      C,A             ;DATA IN C F
OR OUTSER
C430 CD51C1    09315         CALL    URST
C433 CD4AC4    09320         CALL    OUTSER
C436 C9        09330         RET
C437 00        09400 INPC    NOP             ;GET SERIAL INPUT
C438 CD6AC4    09405         CALL    INSER           ;GET SERIAL
INPUT
C43B B7        09410         OR      A
C43C 28F9      09420         JR      Z,INPC          ;WAIT FOR EN
TRY
C43E 57        09430         LD      D,A
C43F DE20      09435         SBC     A,32
C441 3005      09440         JR      NC,RET2
C443 7A        09445         LD      A,D
C444 FE0D      09450         CP      0DH             ;C.R.
C446 20EF      09455         JR      NZ,INPC
C448 7A        09490 RET2    LD      A,D                     ;RET
URN ON NOT # ENTRY
C449 C9        09495         RET
C44A DB09      09500 OUTSER  IN      A,(CTRL)        ;GET UART ST
ATUS REG.
C44C E601      09502         AND     1
C44E 28FA      09510         JR      Z,OUTSER        ;IF NOT, WAI
T
C450 79        09515         LD      A,C             ;DATA IN C R
EG.
C451 D308      09520         OUT     (DATA),A        ;OUTPUT BYTE

C453 C9        09525         RET
C454 F5        09600 CR      PUSH    AF
C455 3E0D      09605         LD      A,0DH           ;CR
C457 CD2BC4    09610         CALL    OUTDRV
C45A 3E0A      09620         LD      A,0AH           ;L.F.
C45C CD2BC4    09630         CALL    OUTDRV
C45F F1        09635         POP     AF
C460 C9        09640         RET
C461 3E0D      09650 CR1     LD      A,0DH           ;C.R.
C463 CD2BC4    09660         CALL    OUTDRV
               09665
C466 C9        09670         RET
C467 CD51C1    09690 URST1   CALL    URST
C46A CDBCC6    09700 INSER   CALL    SCK
C46D DB09      09701         IN      A,(CTRL)        ;GET UART ST
ATUS REG.
C46F E620      09702         AND     20H
C471 FE14      09703         CP      20
C473 28F2      09704         JR      Z,URST1
C475 DB09      09705         IN      A,(CTRL)
C477 E602      09708         AND     2H
C479 FE02      09709         CP      2
C47B 20ED      09710         JR      NZ,INSER        ;IF NOT REPE
AT
[illegible]
```

```
C4B0 DB08   09715        IN      A,(DATA)        ;GET DATUM
C4B2 E67F   09717        AND     7FH
C4B4 C9     09720        RET
C4B5 CDBCC6 09780 INCS   CALL    SCK
C4B8 CD8FC4 09782        CALL    INSNW           ;***(INSNW)
FOR SERIAL
C4BB C9     09790        RET
C4BC CD51C1 09795 URS12  CALL    URS1
C4BF DB09   09800 INSNW  IN      A,(CTRL)        ;GET UART ST
ATUS REG.
C4C1 E620   09802        AND     20H
C4C3 FE14   09804        CP      20
C4C5 28F5   09806        JR      Z,URS12
C4C7 DB09   09808        IN      A,(CTRL)
C4C9 E602   09810        AND     2               ;IS RCVD BYT
E AVAIL.?
C4CB FE02   09815        CP      2
C4CD 2008   09820        JR      NZ,NOBYTE
C4CF DB08   09830        IN      A,(DATA)        ;GET DATUM
C4D1 E67F   09831        AND     7FH
C4D3 B7     09832        OR      A
C4D4 28E6   09834        JR      Z,URST2
C4D6 C9     09840        RET
C4D7 3EFF   09850 NOBYTE LD      A,0FFH          ;NO CHARC. B
YTE
C4D9 C9     09860        RET
C4DA CDA5C5 09905 UTDAT  CALL    INIT4
C4DD 0E0F   09906        LD      C,15
C4DF AF     09907        XOR     A
C4E0 ED79   09908        OUT     (C),A           ;TURN OFF SB
C RX AND TX
C4E2 0E0C   09909        LD      C,12
C4E4 2100CA 09910        LD      HL,CONSEL       ;I/O DATA AR
RAY
C4E7 0611   09915        LD      B,11H           ;ARRAY SIZE
+1
C4E9 EDB3   09920        OTIR                    ;DMA
C4EB CDCFC5 09925        CALL    INIT5
C4EE C9     09926        RET
C4EF 3E40   10000 IUART  LD      A,40H           ;RESET UART
C4F1 D309   10010        OUT     (CTRL),A
C4F3 3E4D   10015        LD      A,04DH          ;8BIT 1STOP
300BAUD
C4F5 D309   10020        OUT     (CTRL),A
C4F7 3E15   10025        LD      A,15H
C4F9 D309   10035        OUT     (CTRL),A                ;RES
ET ERROR FLAGS
C4FB 0E07   10040        LD      C,7
C4FD 3E36   10045        LD      A,36H
C4FF ED79   10050        OUT     (C),A
C501 0E04   10055        LD      C,4
C503 3ED0   10060        LD      A,0D0H
C505 ED79   10065        OUT     (C),A
C507 AF     10066        XOR     A
C508 ED79   10067        OUT     (C),A
C50A C9     10068        RET
0009        10100 CTRL   EQU     09H
0008        10110 DATA   EQU     08H
C800        11000 BARRAY EQU     0C800H          ;BAUD ASSOC.
 WITH BRS
CA00        11015 CONSEL EQU     0CA00H          ;COMP/BR LIN
E
CA20        11020 OUTDAT EQU     0CA20H
CA30        11025 TEMP   EQU     0CA30H          ;TEMP NAME O
N SIGNON
```

```
CB01            11510 CNTR1    EQU     0CB01H          ;BH# FOR COM
P LINE (ALLO)
CB02            11520 TSEL     EQU     0CB02H
CB03            11530 TSCAN    EQU     0CB03H
CB04            11540 DUM      EQU     0CB04H
CB05            11550 FLAG2    EQU     0CB05H
CB06            11560 VAR1     EQU     0CB06H
CB07            11570 DUM3     EQU     0CB07H
CB80            11700 N1EMP    EQU     0CB80H
CB82            11710 N1EMP1   EQU     0CB82H
CB84            11720 DUM1     EQU     0CB84H
CB86            11730 DUM2     EQU     0CB86H
CB88            11735 SCAN2    EQU     0CB88H          ;SCAN COUNT
FOR .25 SEC DELAY
CB8A            11740 N1EMP2   EQU     0CB8AH          ;INPC PASS T
HRU CNT.
CB90            11750 H23NT    EQU     0CB90H  ;
CB8F            11751 FH23     EQU     H23NT-1
C4DB 0000       11900 ZERO     DEFW    0               ;ZERO
C4DD 00FE       11902 SMAX     DEFW    0FE00H          ;# OF SCANS
FOR '#' (.25 SEC.)
C4DF 00         12000 ALLO     NOP             ;ALLOCATION SCREEN P
RINT
C4E0 DD23       12004          INC     IX
C4E2 CD1EC4     12005          CALL    OUTP
C4E5 3A8FCB     12006          LD      A,(FH23)
C4E8 B7         12007          OR      A
C4E9 200D       12008          JR      NZ,AH23
C4EB CD09C5     12030 ALL1     CALL    IXIY
C4EE 21E4C2     12035          LD      HL,SP1
C4F1 CD1EC4     12040          CALL    OUTP
C4F4 CD54C5     12045          CALL    BHD
C4F7 C9         12055          RET
C4F8 DD7E00     12060 AH23     LD      A,(IX)
C4FB FE0B       12065          CP      11
C4FD 28EC       12070          JR      Z,ALL1
C4FF 2A90CB     12075          LD      HL,(H23NT)
C502 CD1EC4     12080          CALL    OUTP
C505 CD54C4     12085          CALL    CR
C508 C9         12090          RET
                12095 ;
                12100 ;
                12105 ;
C509 00         12500 IXIY     NOP
C50A 2100C8     12510          LD      HL,NARRAY
C50D 1600       12520          LD      D,0
C50F DD7E00     12530          LD      A,(IX)
C512 3201CB     12533          LD      (CNTR1),A               ;BH#
 FOR COMP LIN
C515 5F         12535          LD      E,A
C516 CB23       12537          SLA     E
C518 CB12       12538          RL      D
C51A CB23       12540          SLA     E
C51C CB12       12545          RL      D
C51E CB23       12550          SLA     E
C520 CB12       12555          RL      D
C522 CB23       12560          SLA     E
C524 CB12       12565          RL      D
C526 CB23       12570          SLA     E
C528 CB12       12575          RL      D
C52A 19         12580          ADD     HL,DE
C52B 2282CB     12585          LD      (N1EMP1),HL
C52E 47         12586          LD      B,A
C52F 3A00CB     12587          LD      A,(CNTR)
C532 B8         12588          CP      B
C533 [illegible]
```

```
C537 C9          12591          RET
C538 D630        12700 BHD1     SBC     A,48
C53A 47          12710          LD      B,A
C53B 3A00CB      12720          LD      A,(CNTR)
C53E B8          12730          CP      B
C53F 21E4C2      12735          LD      HL,SP1
C542 2003        12740          JR      NZ,BHD2
C544 21EAC2      12750          LD      HL,SP2
C547 CD1EC4      12760 BHD2     CALL    OUTP
C54A 2A02CB      12770          LD      HL,(N1EMP1)
C54D CD1EC4      12780          CALL    OUTP
C550 CD54C4      12790          CALL    CR
C553 C9          12800          RET
C554 21E1C2      12900 BHD      LD      HL,BH
C557 CD1EC4      12905          CALL    OUTP
C55A 3A01CB      12907          LD      A,(CNTR1)
C55D C630        12908          ADD     A,48
C55F CD2BC4      12910          CALL    OUTDRV
C562 18D4        12911          JR      BHD1
C564 00          13000 INIT     NOP
C565 3E0F        13005          LD      A,0FH
C567 DD2100CB    13010          LD      IX,NARRAY
C56B FD2100CA    13015          LD      IY,CONSEL
C56F 012000      13020          LD      BC,20H
C572 00          13025 INIT1    NOP
C573 DD360000    13030          LD      (IX),00H
C577 FD360000    13035          LD      (IY),00H
C57B DD09        13040          ADD     IX,BC
C57D FD23        13045          INC     IY
C57F FE00        13050          CP      0
C581 2803        13060          JR      Z,FINIT
C583 3D          13070          DEC     A
C584 18EC        13080          JR      INIT1
C586 00          13090 FINIT    NOP
                 13140 ;*** MOVE NAME BHO TO F800H
C587 21F1C2      13150          LD      HL,BH0H
C58A 1100CB      13160          LD      DE,NARRAY
C58D 010E00      13170          LD      BC,0EH
C590 EDB0        13180          LDIR
C592 C9          13190          RET
C593 3E00        13200 INIT2    LD      A,0
C595 DD2130CA    13210          LD      IX,NTARY        ;ZERO TEMP.
NAME ARRAY
C599 0E20        13220          LD      C,20H
C59B DD7700      13230 INIT3    LD      (IX),A
C59E 0D          13240          DEC     C
C59F DD23        13250          INC     IX
C5A1 B9          13260          CP      C
C5A2 20F7        13270          JR      NZ,INIT3
C5A4 C9          13280          RET
C5A5 DD2100CA    13400 INIT4    LD      IX,CONSEL
C5A9 AF          13410          XOR     A
C5AA 3206CB      13411          LD      (VAR1),A
C5AD 4F          13412          LD      C,A
C5AE DD7700      13415          LD      (IX),A
C5B1 0C          13416 INIT5    INC     C
C5B2 DD23        13421          INC     IX
C5B4 3A06CB      13422          LD      A,(VAR1)
C5B7 C610        13423          ADD     A,10H
C5B9 3206CB      13424          LD      (VAR1),A
C5BC DD7E00      13426          LD      A,(IX)
C5BF E60F        13427          AND     0FH             ;ZERO ADDR.
FIELD
C5C1 47          13428          LD      B,A
```

```
C5C5 80          13430           ADD     A,B
C5C6 DD7700      13431           LD      (IX),A
C5C9 79          13432           LD      A,C
C5CA FE0F        13440           CP      0FH
C5CC 20E3        13450           JR      NZ,IN11L
C5CE C9          13460           RET
C5CF DD2100CA    13500 IN115     LD      IX,COMSEL
C5D3 AF          13501           XOR     A
C5D4 4F          13502           LD      C,A
C5D5 0C          13503 IN1T5L    INC     C
C5D6 DD23        13504           INC     IX
C5D8 DD7E00      13505           LD      A,(IX)
C5DB E60F        13506           AND     0FH             ;ZERO ADDR.
FIELD
C5DD DD7700      13507           LD      (IX),A
C5E0 3E0F        13508           LD      A,0FH
C5E2 B9          13509           CP      C
C5E3 20F0        13510           JR      NZ,IN1T5L
C5E5 C9          13511           RET
C5E6 21E4C2      14900 SPACE     LD      HL,SP1
C5E9 CD1EC4      14910           CALL    OUTP
C5EC C9          14920           RET
C5ED 20          15000 C0        DEFM    '  DISCONNECT    '
C5FD 00          15002           DEFB    00H
C5FE 20          15005 C1        DEFM    '  A PROC. - TT1 '
C60E 00          15007           DEFB    00H
C60F 20          15010 C2        DEFM    '  B PROC. - TT1 '
C61F 00          15012           DEFB    00H
C620 20          15015 C3        DEFM    '  C PROC. - TT1 '
C630 00          15017           DEFB    00H
C631 20          15020 C4        DEFM    '  GPC     - TT1 '
C641 00          15022           DEFB    00H
C642 20          15030 C5        DEFM    '  CMS LSI - TT0 '
C652 00          15032           DEFB    00H
C653 20          15035 C6        DEFM    '  HAR 11/23 TT0 '
C663 00          15037           DEFB    00H
C664 20          15040 C7        DEFM    '  M111 RS - TT0 '
C674 00          15045           DEFB    00H
C675 20          15050 C8        DEFM    '  11/70   - TT1 '
C685 00          15055           DEFB    00H
C686 20          15060 C9        DEFM    '  11/70   - TT2 '
C696 00          15065           DEFB    00H
C697 20          15070           DEFM    '  A PROC TO H23 '
C6A7 00          15080           DEFB    00H
C6A8 00          16000 DSEL      NOP             ;DELSECT PRESENT BH#

C6A9 3A00CB      16003           LD      A,(CNTR)
C6AC 00          16005 DSELA     NOP
C6AD 2100CA      16010           LD      HL,COMSEL
C6B0 011000      16020           LD      BC,10H
C6B3 E60F        16035           AND     0FH             ;MASK OFF UP
PER ADDR. BITS
C6B5 EDB1        16040           CPIR
C6B7 C0          16070           RET     NZ
C6B8 2B          16075           DEC     HL
                 16077
                 16078
C6B9 3600        16080           LD      (HL),00H
C6BB C9          16090           RET
                 16500 ; ROUTINE TO KICK BACK TO SCAN CODE IF TERMI
NAL SIGNED ON BUT INACTIVE
C6BC E5          16510 SCN       PUSH    HL
C6BD 2A8AC8      16512           LD      HL,(TEMP2)      ;LOCAL CNTR
STORAGE
C6C0 110100      16520           LD      DE,01
```

```
C6C4 DA1FC1   16540          JP      C,[illegible]A1          ;EXIT THRU N
O ACTIVITY
C6C7 228ACB   16550          LD      (NTEMP2),HL      ;STORE INCRE
MENTED CNT.
C6CA E1       16552          POP     HL
C6CB C9       16560          RET
C6CC E5       16600 SCKEX    PUSH    HL
C6CD 210000   16602          LD      HL,00H
C6D0 228ACB   16610          LD      (NTEMP2),HL
C6D3 E1       16612          POP     HL
C6D4 C9       16620          RET
C6D5 00       17000 SCAN     NOP              ;MUX SCANNING SUB. F
OR 3 '#'S
C6D6 AF       17002          XOR     A
C6D7 3203CB   17004          LD      (TSCAN),A        ;ZERO SCAN C
OUNT
C6DA 2ADBC4   17006          LD      HL,(ZERO)
C6DD 2288CB   17008          LD      (SCAN2),HL
C6E0 CD85C4   17010 SCAN1    CALL    INCS
C6E3 F5       17013          PUSH    AF
C6E4 CD0BC7   17015          CALL    MUX
C6E7 F1       17017          POP     AF
C6E8 06FF     17020          LD      B,0FFH  ; INSNW RET FOR NO C
HARC.
C6EA B8       17030          CP      B        ;SEE IF BYTE AVAIL
C6EB 28F3     17040          JR      Z,SCAN1 ;LOOP IF NO BYTE
C6ED 0623     17050          LD      B,35    ;ASCII FOR #
C6EF B8       17060          CP      B
C6F0 2802     17070          JR      Z,SCOK  ;A HAD #
C6F2 18E1     17080          JR      SCAN
C6F4 3A03CB   17090 SCOK     LD      A,(TSCAN)
C6F7 3C       17100          INC     A
C6F8 3203CB   17110          LD      (TSCAN),A
C6FB 0603     17120          LD      B,3
C6FD B8       17130          CP      B        ;3 '#'S ?
C6FE 20E0     17140          JR      NZ,SCAN1
C700 CD71C7   17141          CALL    DSELC
C703 0E0F     17142          LD      C,15    ;SET SBC TX & RX LIN
E TO BH# (CNTR)
C705 3A00CB   17144          LD      A,(CNTR)
C708 ED79     17146          OUT     (C),A
C70A C9       17150          RET
C70B 3A03CB   18000 MUX      LD      A,(TSCAN)
C70E FE00     18002          CP      0
C710 200E     18004          JR      NZ,RET5
C712 2A88CB   18008          LD      HL,(SCAN2)
C715 23       18010          INC     HL
C716 2288CB   18012          LD      (SCAN2),HL
C719 ED5BDDC4 18015          LD      DE,(SMAX)
C71D 19       18020          ADD     HL,DE
C71E 3801     18030          JR      C,MUXCH
C720 C9       18040 RET5     RET
C721 CDCCC6   18045 MUXCH    CALL    SCKEX
C724 CDBFC4   18050          CALL    IOART
C727 3A00CB   18052          LD      A,(CNTR)
C72A 3C       18060          INC     A
C72B FE10     18070          CP      10H              ;16 LINE SCA
N
C72D 2002     18080          JR      NZ,MUX0
C72F 3E01     18090          LD      A,1
C731 3200CB   18100 MUX0     LD      (CNTR),A
C734 0E0E     18105          LD      C,14
C736 ED79     18110          OUT     (C),A
C738 2ADFC4   18115          LD      HL,(ZERO)
C73B 2288CB   18117          LD      (SCAN2),HL
```

```
C73F AF          19000 BOOT    XOR     A
C740 3200CB      19010         LD      (CNTR),A
C743 CD21C7      19040 BOL     CALL    MUXCH
C746 0E0F        19042         LD      C,15
C748 ED79        19044         OUT     (C),A
C74A CD54C4      19046         CALL    CR
C74D CD54C4      19048         CALL    CR
C750 2129C2      19050         LD      HL,MSG0
C753 CD1EC4      19052         CALL    OUTP
C756 21A7C3      19058         LD      HL,BOOT1
C759 CD1EC4      19060         CALL    OUTP
C75C CD54C4      19061         CALL    CR
C75F 3A00CB      19062         LD      A,(CNTR)
C762 FE0F        19063         CP      0FH             ;MAX# BR LIN
ES
C764 2802        19064         JR      Z,BOOTR
C766 18DB        19065         JR      BOL
C768 3E01        19070 BOOTR   LD      A,1
C76A CD31C7      19080         CALL    MUX0
C76D CD8FC7      19085         CALL    BEEP
C770 C9          19090         RET
C771 2100CA      19200 DSELC   LD      HL,COMSEL
C774 011000      19210         LD      BC,10H
C777 3A00CB      19220         LD      A,(CNTR)
C77A E60F        19230         AND     0FH
C77C EDB1        19240         CPIR
C77E 200E        19250         JR      NZ,DSEX1
C780 2B          19260         DEC     HL
C781 7D          19265         LD      A,L
C782 CB27        19270         SLA     A
C784 CB27        19275         SLA     A
C786 CB27        19280         SLA     A
C788 CB27        19285         SLA     A
C78A 0E0C        19290         LD      C,12
C78C ED79        19300         OUT     (C),A
C78E C9          19310 DSEX1   RET
C78F AF          19500 BEEP    XOR     A
C790 3200CB      19510         LD      (CNTR),A
C793 3204CB      19515         LD      (DUM),A
C796 0E07        19530 BP1     LD      C,7
C798 CD4AC4      19540         CALL    OUTSER
C79B CDB2C7      19555         CALL    DELAY2
C79E CD21C7      19565         CALL    MUXCH
C7A1 0E0F        19570         LD      C,15
C7A3 ED79        19575         OUT     (C),A
C7A5 3A04CB      19580         LD      A,(DUM)
C7A8 0610        19590         LD      B,10H
C7AA 3C          19600         INC     A
C7AB 3204CB      19605         LD      (DUM),A
C7AE B8          19610         CP      B
C7AF 20E5        19620         JR      NZ,BP1
C7B1 C9          19625         RET
C7B2 AF          19700 DELAY2  XOR     A
C7B3 3284CB      19705         LD      (DUM1),A
C7B6 CD56C1      19710 DELL    CALL    DELAY1
C7B9 3A84CB      19715         LD      A,(DUM1)
C7BC 3C          19720         INC     A
C7BD 3284CB      19725         LD      (DUM1),A
C7C0 FE20        19730         CP      20H
C7C2 20F2        19735         JR      NZ,DELL
C7C4 C9          19740         RET
0000             20000         END
00000 TOTAL ERRORS
02351  TEXT AREA BYTES LEFT
```

```
ALL1    C4EB 12030   12070
ALLO    C4DF 12000   02100
ALLO1   C0DE 02400   02090
ALLOC   C0A2 02000   01020
ALLOC1  C0CH 02090   02160
BEEP    C78F 19500   19085
BH      C2E1 08210   00110 12900
BHOM    C2F1 08310   13150
BM14    0008 00090   01912 01979
BMD     C554 12900   12045
BMD1    C538 12700   12911
BMD2    C547 12760   12740
BOL     C743 19040   19065
BOOT    C73F 19000   00220
BOOT1   C3A7 08805   19058
BOO1R   C768 19070   19064
BP1     C796 19530   19620
C0      C5ED 15000   02006
C1      C5FE 15005   02060
C2      C60F 15010
C3      C620 15015
C4      C631 15020
C5      C642 15030
C6      C653 15035
C7      C664 15040
C8      C675 15050
C9      C686 15060
CIO     0046 00120
CKFL    C189 05300   04005
CNTR    CB00 11500   01915 01974 01982 01991 04080 07060 12587
                     12720 16003 17144 18052 18100 19010 19062
                     19220 19510
CNTR1   CB01 11510   12533 12907
COMSEL  CA00 11015   02065 04050 09910 13015 13400 13500 16010
                     19200
CR      C454 09600   01065 02030 02280 03000 03050 04100 07010
                     07020 07070 07153 07170 08004 12085 12790
                     19046 19048 19061
CR1     C461 09650
CTRL    0009 10100   05010 09500 09701 09705 09800 09808 10010
                     10020 10035
D2      C15A 05110   05140
DATA    0008 10110   09520 09715 09830
DELAY1  C156 05100   09305 19710
DELAY2  C782 19700   19555
DELETE  C164 05200   07124
DELL    C7B6 19710   19735
DSEL    C6A8 16000   01985 04040
DSEL14  C08B 01971   01107 01109
DSELA   C6AC 16005   01920
DSELC   C771 19200   17141
DSEX1   C7BE 19310   19250
DSP     0033 00100
DUM     CB04 11540   02004 02400 02420 05222 05262 19515 19580
                     19605
DUM1    C684 11720   05223 05264 08003 08010 19705 19715 19725
DUM2    CB86 11730   02008 02460 02480 05224 05266 08002 08011
DUM3    CB07 11570   02080 02110 02130
FB23    CB8E 11751   01130 01155 01964 01973 12006
FIN11   C586 13090   13060
FLAG2   CB05 11550   03084 05300 06060 07016
H23B1   CB90 11750   01966 11751 12075
HEL     C1E9 07150   07130
INCS    C4B5 09780   17010
```

```
INIT1   C572 13025   13080
INIT2   C593 13200   00170 06045 07012
INIT3   C59B 13230   13270
INIT4   C5A5 13400   09905
INITS   C5CF 13500   09925
INITSL  C5D5 13503   13510
INITL   C5B1 13416   13450
INPC    C437 09400   01050 09420 09455
INPC1   C411 09050   07120 09060 09075
INPL    C33A 08410   01030
INSER   C46A 09700   09050 09405 09710
INSNW   C48F 09800   09782
IUART   C4EF 10000   00200 00210 18050
IXIY    C509 12500   12030
KBD     0028 00110
MAIN    C022 01010   01124 01128 01900 01970
MAIN1   C062 01155   01145
MSG0    C229 08020   07030 19050
MSG01   C25B 08040   02040
MSG1    C2B0 08200   02020
MSG2    C2FF 08330
MSG3    C35D 08500   03010
MSG4    C37H 08520   03060 03320
MSG5    C38A 08600   03302
MUX     C70B 18000   17015
MUX0    C731 18100   18080 19380
MUXCH   C721 18045   18030 19040 19565
NALP    C1DA 07120   07126 07145 08015
NARRAY  C800 11000   12510 13010 13160
NUAT    C11F 03300   16540
NUBYTE  C4A7 09850   09820
NTANY   CA30 11025   01965 06000 07118 07160 08006 13210
NTEMP   CB80 11700   03030 06010 12590
NTEMP1  CB82 11710   12585 12770
NTEMP2  CB8A 11740   16512 16550 16610
OTDAT   C4AA 09905   00190 00215 03080 03332
OUTDAT  CA20 11020
OUTDRV  C42B 09300   01060 02430 03210 05244 05250 07065 07123
                     09130 09610 09630 09660 12910
OUTP    C41E 09100   01040 02025 02050 02490 03020 03040 03070
                     03310 03330 07040 07055 07080 07158 07165
                     08008 09150 12005 12040 12080 12760 12780
                     12905 14910 19052 19060
OUTSER  C44A 09500   09320 09510 19540
REPTN   C20F 08002   07134
RET     C42A 09200   09120
RET2    C448 09490   09440
RET4    C1H8 05270   05210
RET5    C720 18040   18064
SCAN    C6D5 17000   00305 17080
SCAN1   C6E0 17010   17040 17140
SCAN2   CHH8 11735   17008 18008 18012 18117
SCR     C6HC 16510   09700 09780
SCREX   C6CC 16600   00300 09712 18045
SCUR    C6F4 17090   17070
SELA14  C068 01912   01105
SETL    C132 04000   01960
SETL1   C135 04007   05330 05360
SETLR   C2EC 08300
SGNON   C3D3 08905   07050
SGNON1  C3F7 08915   07075
SGNON2  C40A 08935   07155
SIGNON  C1AD 07010   00310 07137
SMAX    C4DD 11902   18015
SOFF    C0F9 03000   01120
```

```
SPACE   C5E6 14900   02440
START   C019 00300   03090 03340
START1  C000 00152
SWITCH  C196 06000   05340
TSCAN   CB03 11530   17004 17090 17110 18000
TSEL    CB02 11520   01090 01940 04060
URST    C151 05000   09315 09690 09795
URST1   C467 09690   09704
URST2   C48C 09795   09806 09834
VAR1    CB06 11560   13411 13422 13424 13429
ZERO    C4DB 11900   17006 18115
```

APPENDIX H
FTCRBN

H

```
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:57:53    DB0:[212,10]FTCRBN.LST;1    01234567890123456789
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:57:53    DB0:[212,10]FTCRBN.LST;1    01234567890123456789
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:57:53    DB0:[212,10]FTCRBN.LST;1    01234567890123456789
```

```
FFFFFFFFFF   TTTTTTTTTT     CCCCCCCC   RRRRRRRR     BBBBBBBB     NN      NN
FFFFFFFFFF   TTTTTTTTTT     CCCCCCCC   RRRRRRRR     BBBBBBBB     NN      NN
FF              TT       CC           RR      RR   BB      BB   NN      NN
FF              TT       CC           RR      RR   BB      BB   NN      NN
FF              TT       CC           RR      RR   BB      BB   NNNN    NN
FF              TT       CC           RR      RR   BB      BB   NNNN    NN
FFFFFFFF        TT       CC           RRRRRRRR     BBBBBBBB     NN  NN  NN
FFFFFFFF        TT       CC           RRRRRRRR     BBBBBBBB     NN  NN  NN
FF              TT       CC           RR  RR       BB      BB   NN    NNNN
FF              TT       CC           RR  RR       BB      BB   NN    NNNN
FF              TT       CC           RR    RR     BB      BB   NN      NN      ....
FF              TT       CC           RR    RR     BB      BB   NN      NN      ....
FF              TT         CCCCCCCC   RR      RR   BBBBBBBB     NN      NN      ....
FF              TT         CCCCCCCC   RR      RR   BBBBBBBB     NN      NN      ....
```

H-2

```
LL            SSSSSSSS   TTTTTTTTTT      ;;;;        11
LL            SSSSSSSS   TTTTTTTTTT      ;;;;        11
LL          SS              TT          ;;;;      1111
LL          SS              TT          ;;;;      1111
LL          SS              TT                      11
LL          SS              TT                      11
LL            SSSSSS        TT          ;;;;        11
LL            SSSSSS        TT          ;;;;        11
LL                  SS      TT          ;;;;        11
LL                  SS      TT          ;;;;        11
LL                  SS      TT            ;;        11
LL                  SS      TT            ;;        11
LLLLLLLLLL   SSSSSSSS       TT          ;;        111111
LLLLLLLLLL   SSSSSSSS       TT          ;;        111111
```

```
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:57:53    DB0:[212,10]FTCRBN.LST;1    01234567890123456789
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:57:53    DB0:[212,10]FTCRBN.LST;1    01234567890123456789
01234567890123456789    ** RSX-11M V03 **     20-APR-82    15:57:53    DB0:[212,10]FTCRBN.LST;1    01234567890123456789
```

```
C       AUTHOR: BILL MORRIS
C
C       CHANGES BY:             ROBERT B. NORMOYLE
C                               QUEST RESEARCH
C
C       DOCUMENTATION OF CHANGES
C
C       THE UPDATED VERSION OF FTC.FTN CALCULATES THE POWER
C       AND PHASE OF EACH ANTENNAE IN A QUAD FOR A RANGE OF
C       EFFECTIVE RADIATING POWER (ERP) OF 30 DB
C       TO -20 DB.
C
C       THE OPERATOR TYPES IN THE FREQUENCY THAT HE WISHES
C       TO USE, AND THE POSITION OF THE POINT THAT
C       HE WISHES TO GENERATE. ALSO, HE CAN
C       SPECIFY WHETHER HE WANTS THE OUTPUT ON THE CRT,
C       OR BOTH THE CRT AND LINE PRINTER.
C
C       IF HE SPECIFIES THE LINE PRINTER THE PROGRAM
C       WILL PRINT THE SAME INFORMATION THAT IS TYPED
C       ON THE CRT, ALONG WITH THE CHANGE IN ATTENUATION
C       AND PHASE.
C
C       THE DATA THAT IS PRINTED ON THE CRT AND
C       THE LINE PRINTER(IF SPECIFIED) ARE:
C
C               1.ATTENUATION OF EACH ANTENNAE
C               2.PHASE OF SIGNAL IN EACH ANTENNAE
C               3.LOC OF POINT (0,0 IS CENTER OF ARRAY)
C               4.ERP
C               5.POLARIZATION
C               6. FEED: TWTA,GAAS,BOTH,ETC.ETC
C               7. OTHER DATA
C       THE ABOVE SIX ITEMS ARE PRINTED FOR BOTH FEED 1, AND FEED 2
C
C
C       NOTE: THE DATA FILE FOR  YOUR OPERATING FREQUENCY MUST
C               BE IN YOUR DIRECTORY. EXAMPLE:
C                       FTC1680.DAT
C               THIS FILE IS DATA FOR OPERATING AT 1680 MHZ.
C               WHEN THIS PROGRAM ASKS FOR FREQUENCY THEN TYPE  1680
C
0001            LOGICAL*1 RANDOM
0002            COMMON/TCAPRM/AZ,EL,ERP,JFD,RACW,SLCW(2,4),IASN(4),IBPC
0003            COMMON/ALOG2/ALOG2(128)
0004            COMMON OUTFLG,DB2(2,4),DB1(2,4),P2(2,4),P1(2,4)
0005            DO 5 J=1,128
0006    5       ALOG2(J)=ALOG((127.5+FLOAT(J))/
                1       256.)/ALOG(2.)-128.
0007            CALL JTCAI
0008    10      TYPE *,'ENTER 2 TO PRINT & TYPE, 1 TO TYPE, & 0 TO EXIT'
0009            ACCEPT *,OUTFLG
0010            AZ1=0.
0011            EL=0.
C               INITIALIZE MEMORY USED TO DETERMINE DELTA IN ATN & DEG
0012            DO 20 LOOP=1,2
```

```
                DO 30 L2=1,4
                DB2(LOOP,L2)=0.
                P2(LOOP,L2)=0.
      30        CONTINUE
      20        CONTINUE
      C
                IF (OUTFLG.EQ.0) GO TO 90
                TYPE *,'INPUT AZIMUTH, AND ELEVATION'
                ACCEPT *,AZ,EL
                IF(OUTFLG.EQ.2)PRINT 15,EL,AZ
      15        FORMAT('1ELEVATION=',F8.3,5X,'AZIMUTH=',F8.3,///)
                DO 25 KOUNT=30,-20,-1
                ERP=KOUNT
                I=0
                T0=SECNDS(0.)
      60        DO 55 JFD=1,2
                ISTAT=JTCA()
      55        CONTINUE
                I=I+1
                IF(ISTAT.NE.0) TYPE *,AZ,EL,ERP,ISTAT
                IF(I.LT.100) GO TO 60
                TYPE *,'STATUS=',ISTAT,'  TIME=',SECNDS(T0)*10.
                TYPE *,'AZ=',AZ,'EL=',EL
                TYPE *,'ERP=',ERP
                IF(OUTFLG.NE.2) GO TO 80
                I=0
                T0=SECNDS(0.)
      70        DO 65 JFD=1,2
                ISTAT=JTCA()
      65        CONTINUE
                I=I+1
                IF(ISTAT.NE.0) PRINT *,AZ,EL,ERP,ISTAT
                IF (I.LT.100) GO TO 70
                PRINT *,'STATUS=',ISTAT,' TIME=',SECNDS(T0)*10
                PRINT *,'AZ=',AZ,'EL=',EL
                PRINT *,'ERP=',ERP
      80        CONTINUE
                CALL PRSFI
      25        CONTINUE
                GO TO 10
      90        END
```

PROGRAM SECTIONS

NUMBER	NAME	SIZE		ATTRIBUTES
1	$CODE1	001636	463	RW,I,CON,LCL
2	$PDATA	000170	60	RW,D,CON,LCL
3	$IDATA	000062	25	RW,D,CON,LCL
4	$VARS	000026	11	RW,D,CON,LCL
5	$TEMPS	000004	2	RW,D,CON,LCL
6	TCAPNM	000074	30	RW,D,OVR,GBL
7	ALOG2	001000	256	RW,D,OVR,GBL
8	.$$$$.	000204	66	RW,D,OVR,GBL

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
AZ	R*4	6-000000	AZ1	R*4	4-000004	EL	R*4	6-000004	ERP	R*4	6-000010	I	I*2	4-000016
IBPC	I*2	6-000072	ISTAT	I*2	4-000024	J	I*2	4-000002	JFD	I*2	6-000014	KOUNT	I*2	4-000014
LOOP	I*2	4-000010	L2	I*2	4-000012	OUTFLG	R*4	8-000000	RACW	R*4	6-000016	RANDUM	L*1	4-000000
TO	R*4	4-000020												

ARRAYS

NAME	TYPE	ADDRESS	SIZE		DIMENSIONS
ALOG2	R*4	7-000000	001000	256	(128)
DB1	R*4	8-000044	000040	16	(2,4)
DB2	R*4	8-000004	000040	16	(2,4)
IASN	I*2	6-000062	000010	4	(4)
P1	R*4	8-000144	000040	16	(2,4)
P2	R*4	8-000104	000040	16	(2,4)
SLCW	R*4	6-000022	000040	16	(2,4)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
5	**	10	1-000152	15'	3-000000	20	**	25	**
30	**	55	**	60	1-000562	65	**	70	1-001204
80	1-001562	90	1-001624						

FUNCTIONS AND SUBROUTINES REFERENCED

JTCA JTCAI PRSF1 SECNDS SALOG

TOTAL SPACE ALLOCATED = 003442 913

```
                FUNCTION JTCA()
                PARAMETER NTYP=4,NTGT=2,NBPC=4,NSL=8
                COMMON/TCAPRM/AZ,EL,ERP,JFD,RACW,SLCW(2,4),IASN(4),IBPC
                COMMON /PLCAL/ ASNPL(NSL,32,2)
                COMMON/DUETOA/ATFBYA(NSL,NBPC,5,2),PHIBYA(NSL,5),
                1                ACWBYA(NSL,5),RCWBYA(NTGT,5)
                COMMON/OUTLIM/AMPADJ(NBPC,NTGT),AMPSW(NBPC,NTGT),PSMIN(NSL)
                COMMON/DUETOP/ATNBYP(NSL,5),PCWBYP(NSL,5)
                COMMON/NRFLDC/ANFBYF(5)
                DIMENSION JASN(NTYP),ERPD(NTYP)
                COMMON/ALOG2/ALOG2(128)
                DIMENSION ARG(NTYP),JARG(2,NTYP)
                EQUIVALENCE(ARG(1),JARG(1,1))
                COMMON/TCFILE/TCDAT(15)
                LOGICAL*1 TCDAT
                DATA TCDAT/'S','Y',':','F','T','C',
                1          '0','8','1','0',
                2          '.','D','A','T',0/
                IF=0
                QCOL=(AZ+39.375)/1.25
                IQCOL=INT(QCOL)
                IACOL=INT(.5*QCOL)
                XF=QCOL-FLOAT(IQCOL)
                XC=((ANFBYF(4)*XF+ANFBYF(3))*XF+
                1       ANFBYF(2))*XF+ANFBYF(1)
                IF((IQCOL.AND.1).EQ.0) XC=1.-XC
                QROW=(EL+9.375)/1.25
                IQROW=INT(QROW)
                IAROW=INT(.5*QROW)
                YF=QROW-FLOAT(IQROW)
                YC=((ANFBYF(4)*YF+ANFBYF(3))*YF+
                1       ANFBYF(2))*YF+ANFBYF(1)
                IF((IQROW.AND.1).EQ.0) YC=1.-YC
                IASN(1)=32*(IAROW        )+IACOL
                JASN(1)= 8*(IAROW      -2)+IACOL
                IASN(2)=32*(IAROW        )+IQCOL-IACOL
                JASN(2)= 8*(IAROW      -2)+IQCOL-IACOL
                IASN(3)=32*(IQROW-IAROW  )+IACOL
                JASN(3)= 8*(IQROW-IAROW-2)+IACOL
                IASN(4)=32*(IQROW-IAROW  )+IQCOL-IACOL
                JASN(4)= 8*(IQROW-IAROW-2)+IQCOL-IACOL
                ARG(1)=(1.-XC)*(1.-YC)
                ARG(2)=XC*(1.-YC)
                ARG(3)=(1.-XC)*YC
                ARG(4)=XC*YC
        D       TYPE *,'XF,XC,YF,YC=',XF,XC,YF,YC,'ARG='
        D       TYPE *,ARG
                DO 100 IBPC=1,NBPC
        D       TYPE *,'AMPSW(',IBPC,')=',AMPSW(IBPC,JFD),AMPADJ(IBPC,JFD)
                IF(ERP.GT.AMPSW(IBPC,JFD)) GO TO 100
                AR=AMPADJ(IBPC,JFD)-ERP
                GO TO 200
        100     CONTINUE
                IF=IF.OR.32
                GO TO 2000
        200     ISL=4*(JFD-1)
```

```
0049            DO 1000 I=1,NTYP
0050            ERPD(I)=ERP+6.0205999*(.0078125*FLOAT(JARG(1,1).AND.255*128)+
                1                       ALOG2((JARG(1,I).AND.127)+1))
0051            ISL=ISL+1
0052            ATND=ASNPL(ISL,JASN(I)-11,1)-ERPD(I)
0053            PHID=ASNPL(ISL,JASN(I)-11,2)
        D       TYPE *,'ATND,PHID=',ATND,PHID
0054            ATN=ATFBYA(ISL,IBPC,4,1)/(ATFBYA(ISL,IBPC,1,1)+ATND)+
                1       ATFBYA(ISL,IBPC,3,1)*ATND+ATFBYA(ISL,IBPC,2,1)
0055            A=ATN-AR
0056            IF((63.9375-A).GE.0.) GO TO 300
0057            ATND=ATND+(63.9375-A)
0058            A=63.9375
0059    300     PHI=ATFBYA(ISL,IBPC,4,2)/(ATFBYA(ISL,IBPC,1,2)+ATND)+
                1       ATFBYA(ISL,IBPC,3,2)*ATND+ATFBYA(ISL,IBPC,2,2)
        D       TYPE *,'ATN,PHI=',ATN,PHI
0060            PHIT=PHIBYA(ISL,4)/(PHIBYA(ISL,1)+A)+
                1       PHIBYA(ISL,3)*A+PHIBYA(ISL,2)
0061            P=PHID-PHI-PHIT
0062            IF(P.LT.PSMIN(ISL)) P=P+360.
        D       TYPE *,'A,P=',A,P
0063            ATNT=(((ATNBYP(ISL,5)*P+ATNBYP(ISL,4))*P+ATNBYP(ISL,3))*P+
                1       ATNBYP(ISL,2))*P+ATNBYP(ISL,1)
0064            A=A-ATNT
0065            SLCW(1,I)=ACWBYA(ISL,4)/(ACWBYA(ISL,1)+A)+
                1       ACWBYA(ISL,3)*A+ACWBYA(ISL,2)
0066            IF(SLCW(1,I).GT.63.9375) SLCW(1,I)=63.9375
        D       TYPE *,'A,P=',A,P
0067            SLCW(2,I)=(((PCWBYP(ISL,5)*P+PCWBYP(ISL,4))*P+PCWBYP(ISL,3))*P+
                1       PCWBYP(ISL,2))*P+PCWBYP(ISL,1)
0068            IF(SLCW(1,I).LT.0.) IF=IF.OR.1
0069            IF(SLCW(1,I).GE.64.) IF=IF.OR.2
0070            IF(SLCW(2,1).LT.0.) IF=IF.OR.4
0071            IF(SLCW(2,1).GE.360.) IF=IF.OR.8
0072    1000    CONTINUE
        D       TYPE *,'AR=',AR
0073            RACW=RCWBYA(JFD,4)/(RCWBYA(JFD,1)+AR)+
                1       RCWBYA(JFD,3)*AR+RCWBYA(JFD,2)
0074    2000    JTCA=IF
        D       TYPE *,'RACW=',RACW,'ERPD,SLCW='
        D       TYPE *,ERPD,SLCW
0075            CALL TGTOUT(JFD,IASN,IBPC,SLCW,RACW)
0076            RETURN
0077            ENTRY JTCA1
0078            TYPE *,'ENTER FREQ'
0079            ACCEPT 2100,(TCDAT(I),I=7,10)
0080    2100    FORMAT(4A1)
0081            OPEN(UNIT=2,NAME=TCDAT,TYPE='OLD',READONLY,
                1       FORM='UNFORMATTED')
0082            JPOL=0
0083            READ(2)
0084            READ(2)
0085            READ(2)
0086            READ(2)
0087            READ(2)
0088            READ(2) ASNPL,AMPADJ,AMPSW,PSMIN
```

```
0089            READ(2) ATFBYA,PHIBYA,ACWBYA,RCWBYA,ATNBYP,PCWBYP,ANFBYP
0090            CLOSE(UNIT=2)
0091            CALL PGLOUT(JPGL)
0092            RETURN
0093            END
```

PROGRAM SECTIONS

NUMBER	NAME	SIZE		ATTRIBUTES
1	$CODE1	002456	663	RW,I,CON,LCL
2	$PDATA	000030	12	RW,D,CON,LCL
3	$IDATA	000212	69	RW,D,CON,LCL
4	$VARS	000164	58	RW,D,CON,LCL
5	$TEMPS	000010	4	RW,D,CON,LCL
6	TCAPRM	000074	30	RW,D,OVR,GBL
7	PLCAL	004000	1024	RW,D,OVR,GBL
8	DUETOA	003150	820	RW,D,OVR,GBL
9	OUTLIM	000140	48	RW,D,OVR,GBL
10	DUETOP	000500	160	RW,D,OVR,GBL
11	NREDOC	000024	10	RW,D,OVR,GBL
12	ALOG2	001000	256	RW,D,OVR,GBL
13	TCFILE	000020	8	RW,D,OVR,GBL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
JTCA	I*2	1-000000	JTCA1	I*2	1-002022									

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
A	R*4	4-000136	AR	R*4	4-000112	ATN	R*4	4-000132	ATND	R*4	4-000122	ATNT	R*4	4-000156
AZ	R*4	6-000000	EL	R*4	6-000004	ERP	R*4	6-000010	I	I*2	4-000120	IACOL	I*2	4-000060
IAROW	I*2	4-000100	IBPC	I*2	6-000072	IF	I*2	4-000050	IQCOL	I*2	4-000056	IQROW	I*2	4-000076
ISL	I*2	4-000116	JFD	I*2	6-000014	JPOL	I*2	4-000162	P	R*4	4-000152	PHI	R*4	4-000142
PHID	R*4	4-000126	PHIT	R*4	4-000146	QCOL	R*4	4-000052	QROW	R*4	4-000072	RACW	R*4	6-000016
XC	R*4	4-000066	XF	R*4	4-000062	YC	R*4	4-000106	YF	R*4	4-000102			

ARRAYS

NAME	TYPE	ADDRESS	SIZE		DIMENSIONS
ACWBYA	R*4	8-002640	000240	80	(8,5)
ALOG2	R*4	12-000000	001000	256	(128)
AMPADJ	R*4	9-000000	000040	16	(4,2)
AMPSW	R*4	9-000040	000040	16	(4,2)
ANFBYF	R*4	11-000000	000024	10	(5)
ARG	R*4	4-000000	000020	8	(4)
ASNPL	R*4	7-000000	004000	1024	(8,32,2)
ATFBYA	R*4	8-000000	002400	640	(8,4,5,2)
ATNBYP	R*4	10-000000	000240	80	(8,5)
ERPD	R*4	4-000030	000020	8	(4)
IASN	I*2	6-000062	000010	4	(4)
JARG	I*2	4-000000	000020	8	(2,4
JASN	I*2	4-000020	000010	4	(4)
PCWBYP	R*4	10-000240	000240	80	(8,5)

```
 PHIBYA R*4   8-002400  000240    80    (8,5)
 PSMIN  R*4   9-000100  000040    16    (8)
 RCWBYA R*4   8-003100  000050    20    (2,5)
 SLCW   R*4   6-000022  000040    16    (2,4)
 TCDAT  L*1  13-000000  000017     7    (15)
```

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
100	1-000650	200	1-000704	300	1-001224	1000	**	2000	1-001770
2100'	3-000000								

FUNCTIONS AND SUBROUTINES REFERENCED

CLOSS OPENS POLOUT TGTOUT

TOTAL SPACE ALLOCATED = 014264 3162

```
0001            SUBROUTINE TGTOUT(ITGT,IASN,IPTH,SLCW,RACW)
0002            COMMON/SENSFI/IPA(16,2)
0003            DIMENSION IASN(4),SLCW(2,4),IDF(13),ISW(4),I2(2)
0004            INTEGER*4 I4
0005            EQUIVALENCE (I4,I2(1))
0006            DATA IDF /"40000,"40001,"40002,"40003,
                1       "24,5,6,7,8,"31,"32,"33,"34/
0007            DATA ISW /"40,"140,"240,"340/,ISGN/-32768/
0008            I4=1024.*RACW
0009            IPA(9,1TGT)=IDF(9)+(I2(1).AND."177700)
0010            DO 100 I=1,4
0011            IPA(I,1TGT)=IDF(I)+64*IASN(I)
0012            I4=1024.*SLCW(1,I)
0013            IPA(I+4,ITGT)=IDF(I+4)+(I2(1).AND."177700)
0014            I4=182.04444*SLCW(2,I)
0015    100     IPA(I+9,ITGT)=IDF(I+9)+ISW(IPTH)+(I2(1).AND."177400)
0016            CALL SFI(ITGT)
0017            RETURN
0018            ENTRY POLOUT(IPOL)
0019            IF(IPOL.EQ.0) GO TO 210
0020            DO 200 I=1,4
0021    200     IDF(I)=IDF(I).OR.ISGN
0022    210     DO 220 I=1,13
0023    220     IPA(1,1)=IDF(I)
0024            CALL SFI(1)
0025            RETURN
0026            END
```

PROGRAM SECTIONS

NUMBER	NAME	SIZE		ATTRIBUTES
1	$CODE1	000670	220	RW,I,CON,LCL
2	$PDATA	000010	4	RW,D,CON,LCL
3	$IDATA	000034	14	RW,D,CON,LCL
4	$VARS	000052	21	RW,D,CON,LCL
5	$TEMPS	000006	3	RW,D,CON,LCL
6	SERSF1	000100	32	RW,D,OVR,GBL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
POLOUT		1-000450	TGTOUT		1-000000									

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
I	I*2	4-000050	IPOL	I*2	F-000002*	IPTH	I*2	F-000006*	ISGN	I*2	4-000046	ITGT	I*2	F-000002*
I4	I*4	4-000000	RACW	R*4	F-000012*									

ARRAYS

NAME	TYPE	ADDRESS	SIZE		DIMENSIONS
IASW	I*2	F-000004*	000010	4	(4)
IDF	I*2	4-000004	000032	13	(13)
IPA	I*2	6-000000	000100	32	(16,2)
ISW	I*2	4-000036	000010	4	(4)
I2	I*2	4-000000	000004	2	(2)
SLCW	R*4	F-000010*	000040	16	(2,4)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
100	**	200	**	210	1-000532	220	**		

FUNCTIONS AND SUBROUTINES REFERENCED

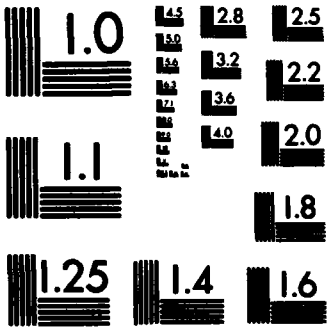
SF1

TOTAL SPACE ALLOCATED = 001114 294

```
0001            SUBROUTINE SFI(ITGT)
0002            INTEGER*4 I4
0003            COMMON/SERSFI/IPA(16,2)
0004            DIMENSION I2(2),SW(8),ICH(4)
0005            COMMON OUTFLG,DB2(2,4),DB1(2,4),P2(2,4),P1(2,4)
0006            EQUIVALENCE (I4,I2(1))
0007            DATA SW/'INH0','BYPS','INH2','GAAS',
                1       'INH4','TWTA','INH6','BOTH'/
0008            DATA ICH/'A','B','C','D'/
0009            RETURN
0010            ENTRY PRSFI
        C       TYPE *,'SFI=',IPA
0011            DO 10 K=1,2
0012            I2(1)=IPA(9,K).AND."177700
0013            TYPE *,'RANGE ATTN=',1./1024.*I4
0014            IF(OUTFLG.EQ.2) PRINT *,'RANGE ATTN=',1./1024.*I4
0015            DO 10 L=1,4
0016            IP='V'
0017            IF((IPA(L,K).AND."100000).EQ.0) IP='H'
0018            IE='E'
0019            IF((IPA(L,K).AND."40000).EQ.0) IE='N'
0020            IR=(IPA(L,K).AND."34000)/2048
0021            IC=(IPA(L,K).AND."3700)/64
0022            I2(1)=IPA(L+4,K).AND."177700
0023            A=1./1024.*I4
0024            I2(1)=IPA(L+9,K).AND."177400
0025            P=180./32768.*I4
0026            ISW4=(IPA(L+9,K).AND."340)/32+1
0027            AUX='FEED'
0028            IF((IPA(L+9,K).AND."20).EQ.0) AUX='AUX '
0029            WRITE(5,100) IR,IC,ICH(L),IP,IE,A,P,SW(ISW4),AUX
0030            IF (OUTFLG.NE.2) GO TO 10
        C       PRINT DATA ON LINE PRINTER WITH DELTA ATN & PHASE
        C
        C       DETERMINE DELTA DB
0031            DB1(K,L)=DB2(K,L)
0032            DB2(K,L)=A
0033            DDB=DB2(K,L)-DB1(K,L)
        C
        C       DETERMINE DELTA PHASE
0034            P1(K,L)=P2(K,L)
0035            P2(K,L)=P
0036            DP=P2(K,L)-P1(K,L)
        C
0037            WRITE(6,110) IR,IC,ICH(L),IP,IE,A,P,SW(ISW4),AUX,DDB,DP
0038    10      CONTINUE
0039            RETURN
0040    100     FORMAT(' ASN ',I1,X,I2,X,A1,X,2A1,X,
                1       F7.4,' DB ',F9.5,' DEG ',A4,X,A4)
0041    110     FORMAT(' ASN ',I1,X,I2,X,A1,X,2A1,X,
                1       F7.4,' DB',X,F9.5,' DEG ',A4,X,A4,7X,F7.4,' DDB',4X,F11.5,
                2       ' DDEG')
0042            END
```

AD-A133 510 DESIGN AND INTEGRATION SUPPORT TO FLIGHT INTERFACE HARDWARE(U) QUEST RESEARCH CORP MCLEAN VA R B NORMOYLE 30 APR 82 QRC-C-4136 N00014-81-C-2499 2/2

UNCLASSIFIED F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

PROGRAM SECTIONS

NUMBER	NAME	SIZE		ATTRIBUTES
1	$CODE1	001354	374	RW,I,CON,LCL
2	$PDATA	000024	10	RW,D,CON,LCL
3	$IDATA	000162	57	RW,D,CON,LCL
4	$VARS	000116	39	RW,D,CON,LCL
6	SERSFI	000100	32	RW,D,OVR,GBL
7	.$$$$.	000204	66	RW,D,OVR,GBL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
PRSFI		1-000014	SFI		1-000000									

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
A	R*4	4-000070	AUX	R*4	4-000102	DDB	R*4	4-000106	DP	R*4	4-000112	IC	I*2	4-000066
IE	I*2	4-000062	IP	I*2	4-000060	IR	I*2	4-000064	ISW4	I*2	4-000100	ITGT	I*2	F-0000028
I4	I*4	4-000000	K	I*2	4-000054	L	I*2	4-000056	OUTFLG	R*4	7-000000	P	R*4	4-000074

H-14

ARRAYS

NAME	TYPE	ADDRESS	SIZE		DIMENSIONS
DB1	R*4	7-000044	000040	16	(2,4)
DB2	R*4	7-000004	000040	16	(2,4)
ICH	I*2	4-000044	000010	4	(4)
IPA	I*2	6-000000	000100	32	(16,2)
I2	I*2	4-000000	000004	2	(2)
P1	R*4	7-000144	000040	16	(2,4)
P2	R*4	7-000104	000040	16	(2,4)
SW	R*4	4-000004	000040	16	(8)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
10	1-001304	100'	3-000000	110'	3-000056				

TOTAL SPACE ALLOCATED = 002204 57H

FTCHBN,FTCHBN=FTCHBN

APPENDIX I
FTC Results

ELEVATION= 0.612 AZIMUTH= 0.125

FTC Results

```
 STATUS=      0  TIME=   7.677083
 AZ=  0.1253000     EL=  0.6120000
 ERP=   30.00000
 RANGE ATTN=   10.62500
ASN 3 15 A ME 53.0625 DB 274.21875 DEG BOTH FEED        53.0625 DDB       274.21875 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED        43.5000 DDB       337.50000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED        12.6875 DDB       170.15625 DDEG
ASN 4 16 D ME  8.1250 DB 208.12500 DEG BOTH FEED         8.1250 DDB       208.12500 DDEG
 RANGE ATTN=   9.500000
ASN 3 15 A ME 46.9375 DB 226.40625 DEG BOTH FEED        46.9375 DDB       226.40625 DDEG
ASN 3 16 B ME 46.5000 DB 253.12500 DEG BOTH FEED        46.5000 DDB       253.12500 DDEG
ASN 4 15 C ME 17.6250 DB 337.50000 DEG BOTH FEED        17.6250 DDB       337.50000 DDEG
ASN 4 16 D ME 11.1250 DB 327.65625 DEG BOTH FEED        11.1250 DDB       327.65625 DDEG
 STATUS=      0  TIME=   7.666667
 AZ=  0.1253000     EL=  0.6120000
 ERP=   29.00000
 RANGE ATTN=   11.56250
ASN 3 15 A ME 53.0625 DB 274.21875 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 16 D ME  8.1875 DB 208.12500 DEG BOTH FEED         0.0625 DDB         0.00000 DDEG
 RANGE ATTN=   10.43750
ASN 3 15 A ME 46.8750 DB 226.40625 DEG BOTH FEED        -0.0625 DDB         0.00000 DDEG
ASN 3 16 B ME 46.5000 DB 253.12500 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 15 C ME 17.6250 DB 337.50000 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 16 D ME 11.1250 DB 327.65625 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
 STATUS=      0  TIME=   7.648438
 AZ=  0.1253000     EL=  0.6120000
 ERP=   28.00000
 RANGE ATTN=   12.56250
ASN 3 15 A ME 53.0625 DB 274.21875 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 16 D ME  8.1250 DB 208.12500 DEG BOTH FEED        -0.0625 DDB         0.00000 DDEG
 RANGE ATTN=   11.37500
ASN 3 15 A ME 46.8125 DB 226.40625 DEG BOTH FEED        -0.0625 DDB         0.00000 DDEG
ASN 3 16 B ME 46.4375 DB 253.12500 DEG BOTH FEED        -0.0625 DDB         0.00000 DDEG
ASN 4 15 C ME 17.6250 DB 337.50000 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 16 D ME 11.1250 DB 327.65625 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
 STATUS=      0  TIME=   7.658854
 AZ=  0.1253000     EL=  0.6120000
 ERP=   27.00000
 RANGE ATTN=   13.56250
ASN 3 15 A ME 53.0625 DB 274.21875 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 16 D ME  8.1250 DB 208.12500 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
 RANGE ATTN=   12.31250
ASN 3 15 A ME 46.7500 DB 226.40625 DEG BOTH FEED        -0.0625 DDB         0.00000 DDEG
ASN 3 16 B ME 46.3750 DB 253.12500 DEG BOTH FEED        -0.0625 DDB         0.00000 DDEG
ASN 4 15 C ME 17.6250 DB 337.50000 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 16 D ME 11.1875 DB 327.65625 DEG BOTH FEED         0.0625 DDB         0.00000 DDEG
 STATUS=      0  TIME=   7.679688
 AZ=  0.1253000     EL=  0.6120000
 ERP=   26.00000
 RANGE ATTN=   14.50000
ASN 3 15 A ME 53.0625 DB 274.21875 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED         0.0000 DDB         0.00000 DDEG
```

```
ASN 3 15 A ME 46.6875 DB 226.40625 DEG BOTH FEED      -0.0625 DDB      0.00000 DDEG
ASN 3 16 B ME 46.3750 DB 253.12500 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 15 C ME 17.6250 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME 11.1875 DB 327.65625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
 STATUS=      0  TIME=   7.051042
 AZ=  0.1253000     EL=  0.6120000
 ERP=   25.00000
 RANGE ATTN=   15.50000
ASN 3 15 A ME 53.0625 DB 274.21875 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME  8.1250 DB 208.12500 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
 RANGE ATTN=   14.18750
ASN 3 15 A ME 46.6875 DB 226.40625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 3 16 B ME 46.3125 DB 253.12500 DEG BOTH FEED      -0.0625 DDB      0.00000 DDEG
ASN 4 15 C ME 17.6250 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME 11.1875 DB 327.65625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
 STATUS=      0  TIME=   7.666667
 AZ=  0.1253000     EL=  0.6120000
 ERP=   24.00000
 RANGE ATTN=   16.43750
ASN 3 15 A ME 53.1250 DB 274.21875 DEG BOTH FEED       0.0625 DDB      0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME  8.1250 DB 208.12500 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
 RANGE ATTN=   15.12500
ASN 3 15 A ME 46.6250 DB 226.40625 DEG BOTH FEED      -0.0625 DDB      0.00000 DDEG
ASN 3 16 B ME 46.2500 DB 253.12500 DEG BOTH FEED      -0.0625 DDB      0.00000 DDEG
ASN 4 15 C ME 17.5625 DB 337.50000 DEG BOTH FEED      -0.0625 DDB      0.00000 DDEG
ASN 4 16 D ME 11.1875 DB 327.65625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
 STATUS=      0  TIME=   8.489583
 AZ=  0.1253000     EL=  0.6120000
 ERP=   23.00000
 RANGE ATTN=   17.43750
ASN 3 15 A ME 53.1250 DB 274.21875 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME  8.1250 DB 208.12500 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
 RANGE ATTN=   16.12500
ASN 3 15 A ME 46.5625 DB 226.40625 DEG BOTH FEED      -0.0625 DDB      0.00000 DDEG
ASN 3 16 B ME 46.1875 DB 251.71875 DEG BOTH FEED      -0.0625 DDB     -1.40625 DDEG
ASN 4 15 C ME 17.5625 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME 11.1875 DB 327.65625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
 STATUS=      0  TIME=   7.848958
 AZ=  0.1253000     EL=  0.6120000
 ERP=   22.00000
 RANGE ATTN=   18.37500
ASN 3 15 A ME 53.1250 DB 274.21875 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME  8.1250 DB 208.12500 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
 RANGE ATTN=   17.06250
ASN 3 15 A ME 46.5000 DB 226.40625 DEG BOTH FEED      -0.0625 DDB      0.00000 DDEG
ASN 3 16 B ME 46.1875 DB 251.71875 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 15 C ME 17.5625 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME 11.1875 DB 326.25000 DEG BOTH FEED       0.0000 DDB     -1.40625 DDEG
 STATUS=      0  TIME=   7.653646
 AZ=  0.1253000     EL=  0.6120000
 ERP=   21.00000
 RANGE ATTN=   19.31250
ASN 3 15 A ME 53.1250 DB 274.21875 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 3 16 B ME 43.5000 DB 337.50000 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 15 C ME 12.6875 DB 170.15625 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
ASN 4 16 D ME  8.1250 DB 208.12500 DEG BOTH FEED       0.0000 DDB      0.00000 DDEG
```

```
ASM 3 16 B HE 46.1250 DB 251.71875 DEG BOTH FEED      -0.0625 DDB       0.00000 DDEG
ASM 4 15 C HE 17.5000 DB 337.50000 DEG BOTH FEED      -0.0625 DDB       0.00000 DDEG
ASM 4 16 D HE 11.1875 DB 326.25000 DEG BOTH FEED       0.0000 DDB       0.00000 DDEG
 STATUS=      0  TIME=    7.964063
 AZ=  0.1253000      EL=  0.0120000
 ERR=   20.00000
 RANGE ATTN=    1.437500
ASM 3 15 A HE 50.9375 DB 279.84375 DEG TWTA FEED      -2.1875 DDB       5.62500 DDEG
ASM 3 16 B HE 42.6750 DB 341.71875 DEG TWTA FEED      -0.6250 DDB       4.21875 DDEG
ASM 4 15 C HE 11.0625 DB 196.87500 DEG TWTA FEED      -1.6250 DDB      26.71875 DDEG
ASM 4 16 D HE  7.9375 DB 174.37500 DEG TWTA FEED      -0.1875 DDB     -33.75000 DDEG
 RANGE ATTN=   18.93750
ASM 3 15 A HE 46.3750 DB 226.40625 DEG BOTH FEED      -0.0625 DDB       0.00000 DDEG
ASM 3 16 B HE 46.0625 DB 251.71875 DEG BOTH FEED      -0.0625 DDB       0.00000 DDEG
ASM 4 15 C HE 17.5000 DB 337.50000 DEG BOTH FEED       0.0000 DDB       0.00000 DDEG
ASM 4 16 D HE 11.1875 DB 326.25000 DEG BOTH FEED       0.0000 DDB       0.00000 DDEG
```

END

FILMED

11-83

DTIC